1969

An approach to the numerical modelling of cumulus-scale motions

Anawalt, Richard Arthur

Monterey, California. U.S. Naval Postgraduate School

AN APPROACH TO THE NUMERICAL MODELLING OF CUMULUS-SCALE MOTIONS

by

Richard Arthur Anawalt

United States
Naval Postgraduate School

THESIS

AN APPROACH TO THE NUMERICAL MODELLING
OF CUMULUS-SCALE MOTIONS

by

Richard Arthur Anawalt

June 1969

AN APPROACH TO THE NUMERICAL MODELLING OF CUMULUS-SCALE MOTIONS

by

Richard Arthur Anawalt
Lieutenant, United States Navy
B.S., Miami University, 1963

Submitted in partial fulfillment of the requirements for the degree of

MASTER OF SCIENCE IN METEOROLOGY

from the

NAVAL POSTGRADUATE SCHOOL
June 1969

ABSTRACT

A numerical model which utilizes the isobaric vorticity equation is developed and applied to cumulus-scale data. The model, together with a modified version of the cumulus convection model of Weinstein and Davis, is applied to data obtained from the National Severe Storms Laboratory in Norman, Oklahoma. The calculations yield real time predictions for height, temperature and relative humidity at seven pressure levels, which are then used as input to the cumulus convection model to obtain vertical profiles of various parameters at specified grid points.

Some results of the calculations are presented along with suggestions for further testing and improvement. The results indicate that further modifications to the approach used are necessary in order to provide more accurate forecasts. Values of the individual terms in the vorticity equation are presented as computed from the observed mesoscale data.

TABLE OF CONTENTS

LIST OF TABLES

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

[illegible]

LIST OF ILLUSTRATIONS

[illegible]

TABLE OF SYMBOLS AND ABBREVIATIONS

a	Threshold value below which conversion from cloud water to hydrometeor water does not take place (gm m^{-3}).
a'	Threshold value below which conversion from cloud water to hydrometeor water does not take place (gm Kg^{-1}).
C_{pv}	Specific heat of water vapor at constant pressure (cal gm^{-1} $^oK^{-1}$).
C_w	Specific heat of liquid water (cal gm^{-1} $^oK^{-1}$).
e_s	Saturation vapor pressure in millibars.
f	Coriolis parameter.
f_m	Mean Coriolis parameter for region studied = f_{35}.
g	Acceleration of gravity (m sec^{-1}).
J	Jacobian.
L_f	Latent heat of fusion (cal gm^{-1}).
L_s	Latent heat of sublimation (cal gm^{-1}).
L_v	Latent heat of vaporization (cal gm^{-1}).
m	Kessler's saturation vapor density (gm m^{-3}).
M	Kessler's precipitation content (gm m^{-3}).
p	Pressure in millibars (mb).
Δq_s	Saturation mixing ratio over water minus saturation mixing ratio over ice (gm gm^{-1}).
Q_c	Cloud water content (gm gm^{-1}).
Q_h	Hydrometeor water content (gm gm^{-1}).
R	Gas constant for dry air $\approx$ 0.28704 (joules gm^{-1} $^oK^{-1}$).
R_h	Relative humidity (%).

t	Time.
T	Temperature.
u	Zonal component of horizontal velocity (m sec^{-1}).
v	Meridional component of horizontal velocity (m sec^{-1}).
V	Terminal velocity of hydrometeors (m sec^{-1}).
$\mathbb{V}$	Horizontal wind vector.
$\mathbb{V}_\chi$	Divergent component of the horizontal wind vector.
$\mathbb{V}_\psi$	Rotational component of the horizontal wind vector.
w	Vertical component of velocity in (x, y, z, t) coordinate system (m sec^{-1}).
z	Height of isobaric surface in meters.
ϵ	Ratio of gas constants $\approx$ .622.
ζ	Vertical component of relative vorticity (sec^{-1}).
ρ	Air density (gm m^{-3}).
Φ	Geopotential = gz.
ψ	Stream function (m^2 sec^{-1}).
ω	Vertical component of velocity in (x, y, p, t) coordinate system = $\frac{dp}{dt}$ (mb hr^{-1} or mb sec^{-1}).
∇	"Del" operator = $\mathbb{i}\frac{\partial}{\partial x} + \mathbb{j}\frac{\partial}{\partial y}$.
∇^2	Laplacian operator = $\frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$.
$\mathbb{\nabla}^2$	Finite difference form of the Laplacian operator.
$\overline{X}$	Average value of any parameter X between two pressure levels.
$(\overline{X})$	Average value of any parameter X with respect to temperature.

ACKNOWLEDGEMENT

The author wishes to thank Dr. R. L. Alberty for his guidance and encouragement throughout the research and during the writing of this thesis, Dr. G. J. Haltiner and Dr. J. D. Mahlman for their invaluable suggestions and advice, and my wife and children for their patience throughout the project.

1. INTRODUCTION

The National Severe Storms Laboratory (NSSL) located at Norman, Oklahoma, has been gathering vast amounts of data during the thunderstorm season for the past several years. Many studies have been conducted with the aid of this data (for example see Barnes, 1968, Newton and Fankhauser, 1964 and Hammond, 1967), however, after an extensive literature search, it appears that no research has been done, or at least reported, involving a numerical modelling approach whereby the distributions of various parameters were analyzed and their values numerically forecast for a future time. Such an approach, if successful, could give distinct advantages in predicting, for example, where hail might be expected, or where the most favorable conditions for tornadoes might be located. In addition, during severe convective activity radiosonde data often terminates near mid-tropospheric levels. Consequently, a model which could reasonably predict the sounding above termination level, when given a complete sounding from the previous observation time, would aid greatly in later research efforts.

Figure 1 shows the region used for this study with a height-field analysis and the reported winds at the nine radiosonde stations in the NSSL upper-air network (see Appendix A for a description of the NSSL upper-air network). Assuming the data to be free of errors, the non-geostrophic nature of the flow on this scale is immediately evident, and the figure emphasizes the need for intensive research into the dynamic mechanisms involved under such circumstances. In an effort to evaluate the possibilities of numerical weather prediction for such small-scale data, this paper uses a numerical prediction model based on the complete isobaric vorticity equation.

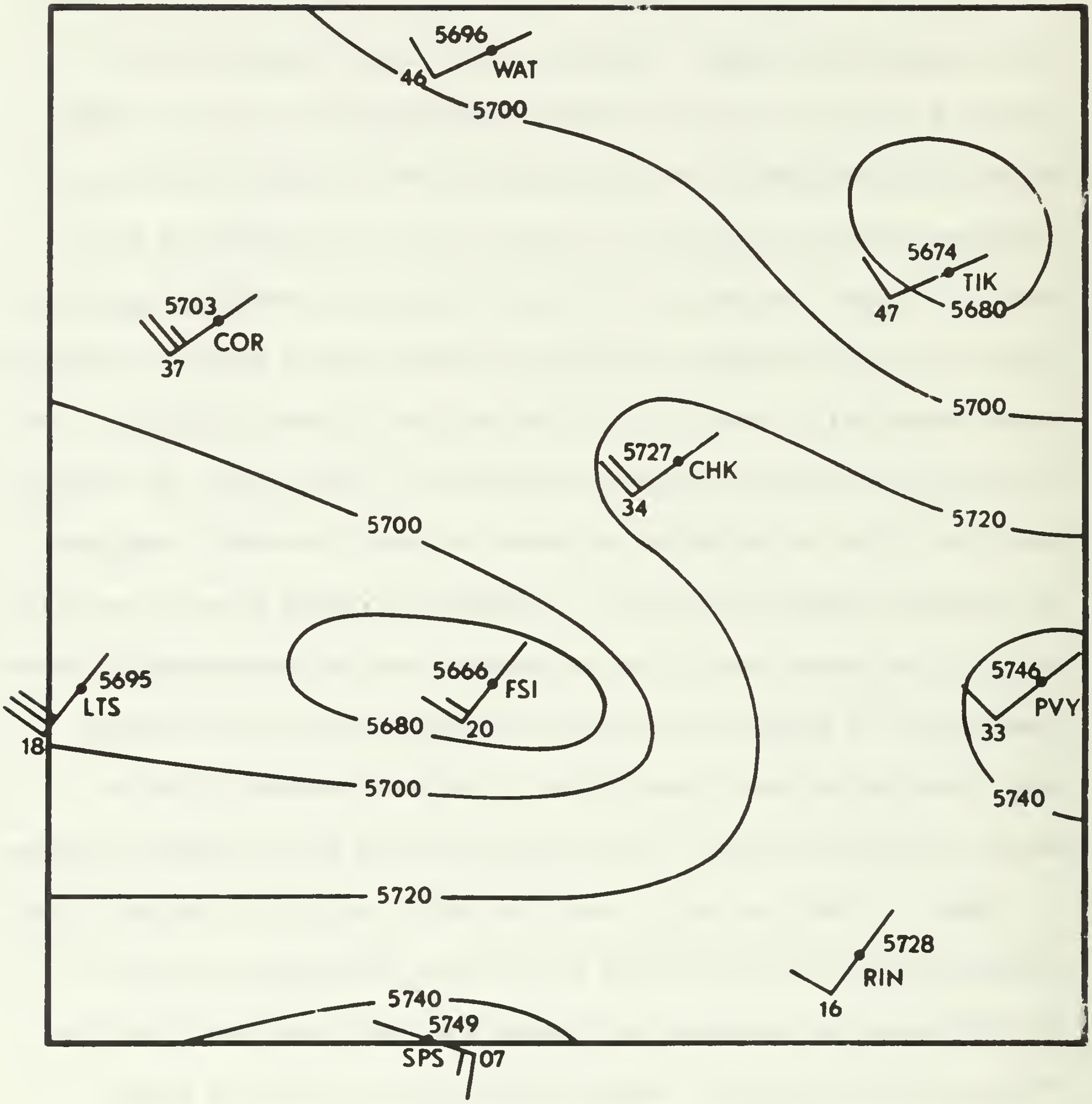

Figure 1: Height analysis with wind directions and speeds (m sec^{-1}) at nine radiosonde stations of NSSL network for the 500-mb level at 1700Z on 30 May 1967 which illustrates the non-geostrophic nature of the data.

The resulting forecasts are then used as input to a cumulus convection model developed by Weinstein and Davis[1] (1967) to provide an aid to severe storm forecasting.

The parameterized cumulus convection model by Weinstein and Davis (1967) calculates cloud top, profiles of vertical velocity, temperature excess (that is, the temperature of a parcel of air in the cloud minus the temperature of a parcel of air in the environment), hydrometeor water content and updraft radius of the cloud as well as rainfall amount and duration. The principles of the above calculations are based on the entrainment principles of Stommel (1947) and the results of Kessler (1967). With a standard radiosonde sounding at mandatory and significant levels beginning at cloud base, the W.D. model transforms the sounding to one which is spaced in equal increments of height. After the user specifies certain boundary conditions such as ice nucleation temperature, radius of the base of the cloud and conversion and collection rates (Kessler, 1967), the model specifies cloud extent. One use of this model is to compare the effect of seeding a given cloud or allowing it to develop naturally. This is accomplished by varying the ice nucleation temperature.

The numerical model presented in this paper predicts heights, temperatures and relative humidities at individual grid points for seven isobaric levels (950 mb, 850 mb, 700 mb, 500 mb, 300 mb, 200 mb, and 150 mb) so that predicted information can be used as input to a modified subroutine form of the W.D. model to yield the vertical

[1]The model developed by Weinstein and Davis has been referred to as the Pennsylvania State model. In this paper, it will be abbreviated as the W.D. model.

profiles previously discussed, under the assumption that cloud base is at 850 mb. However, the present version, as described in this paper, does not provide for vertical coupling of the seven isobaric levels after the first time step.

Predictions were made at 1.5-hour intervals utilizing the IBM OS/360 (MVT versions 15, 16) computer. The 1.5-hour prediction interval was selected to correspond to the times when verification data were available from the NSSL radiosonde network.

The grid used for the computations is a 24 x 24 X-Y grid oriented 000-180, 090-270, having a mean latitude of 35N. The grid distance is 5 nautical miles with the grid arranged so that the nine upper-air stations are either located exactly on a grid point or are so close to one that they may be approximated to be at the nearest grid point. The farthest distance away from a grid point for a particular station was 2 nm. They were assumed to be located at the nearest grid point to eliminate the need for an interpolation scheme in the objective analysis used for initialization of the data as described in Section 4.

2. MODIFICATIONS TO THE W.D. MODEL

While working with and studying the W.D. model, it became apparent that several modifications could be made to it that would provide more accuracy. Although most of the modifications were of a minor nature, three of the modifications are considered to be significant improvements over the original model. These three modifications will be presented in detail while the remainder of the changes will be summarized at the end of this section since they are considered to be self-explanatory.

Kessler (1967) proposed the equation

$$\frac{dM}{dt} = - \frac{dm}{dt} = k_1(m-a) \qquad (2\text{-}1a)$$

to model the conversion of cloud water to hydrometeor water. Here, k_1 is an empirical constant which is equal to .001 sec^{-1} when $m>a$ and is zero for $m<a$. The quantity "a" is defined as a threshold value and refers to the cloud water content. If the cloud water content is greater than this threshold value, then conversion of cloud water to hydrometeor water will occur. If cloud water amounts less than the threshold value are present, then conversion of cloud water to hydrometeor water will not take place. This equation is incorporated into the W.D. model (Weinstein and Davis, 1967) in the form

$$\frac{dQ_h}{dt} = - \frac{dQ_c}{dt} = k_1(Q_c-a') \qquad (2\text{-}1b)$$

where k_1 is defined in a similar manner as it was in (2-1a). Kessler (1967) indicated that measurements have been performed which indicate that cloud water amounts greater than 1 gm m^{-3} are usually associated

with the production of precipitation. Furthermore, he arbitrarily suggested the value of 0.5 gm m^{-3} for the threshold value. Weinstein and Davis (1967) chose a' = 0.5 gm Kg^{-1} for their threshold value.

It is proposed to consider this threshold value to be a function of density. This approach thereby associates a different threshold value with each level by making it a function of height. Using this approach, the conversion of cloud water content to precipitation in the W.D. model has been modified to

$$\frac{dQ_h}{dt} = k_1(Q_c - a/\rho) \qquad (2\text{-}2)$$

where ρ = air density. By defining k_1 as it was in (2-1a), conversion of cloud water to precipitation is modelled to occur when $(Q_c - \frac{a}{\rho}) > 0$.

Computations have been made to determine the variation of the quantity a/ρ. These values show a maximum of approximately .00137 gm gm^{-1} in the upper levels and a minimum value of .00076 gm gm^{-1} near cloud base.

The second significant modification is the inclusion of a linear variation of the latent heats of vaporization (L_v) and fusion (L_f) with temperature. Since

$$L_s = L_v + L_f \qquad (2\text{-}3)$$

and L_s may be considered constant while L_f is a function of temperature, then L_v is also a function of temperature. The values of L_f (List, 1951) are 48.6 cal gm^{-1} at -50C and 79.7 cal gm^{-1} at 0C. The latent heat of vaporization varies from 629.3 cal gm^{-1} at -50C to 597.3 cal gm^{-1} at 0C. Since L_s varies by only 1 cal gm^{-1} from 0C to -40C, its value was taken as constant and equal to 678 cal gm^{-1}. The

variation of only one of the latent heats on the right side of equation (2-3) needs to be computed since the value of the second one can be obtained by subtraction.

In order to approximate the variation of the latent heat of vaporization as a linear function of temperature, it was necessary to determine an optimum value for the difference between the specific heats of water and ice. Integrating the formula

$$\frac{dL_v}{dT} = C_{pv} - C_w \tag{2-4}$$

and applying the mean value theorem for integrals yields

$$L_v = L_{v_o} + (\overline{C_{pv} - C_w})\,(T-T_o). \tag{2-5}$$

Utilizing the values of L_v (List, 1951) for various temperatures (T) and with T_o = -40C, the value of $(\overline{C_{pv} - C_w})$ that satisfies equation (2-5) was computed for each temperature at 10 degree intervals from -40C to +40C. A plot of $(\overline{C_{pv} - C_w})$ versus temperature is shown in Figure 2. Based upon the results it was decided to select the value of -.62340 as an "optimum" value for $(\overline{C_{pv} - C_w})$ and consider it to be a constant. This value is considered to be a representative value between -40C and 0C, and corresponds exactly to the value at -10C.

The final significant modification involves the values used for the constants when computing saturation vapor pressure (Weinstein and Davis, 1967). Integrating the Clausius-Clapeyron equation one obtains the following for the saturation vapor pressure over water:

$$\ln e_s = -\frac{\epsilon\,[L_{vo} - T_o\,(\overline{C_{pv} - C_w})]}{RT} + \frac{\epsilon\,(\overline{C_{pv} - C_w})\ln T}{R}$$

$$+ \frac{\epsilon\,[L_{vo} - T_o\,(\overline{C_{pv} - C_w})]}{RT_o} - \frac{\epsilon\,(\overline{C_{pv} - C_w})\ln T_o}{R} + \ln e_{so}. \tag{2-6}$$

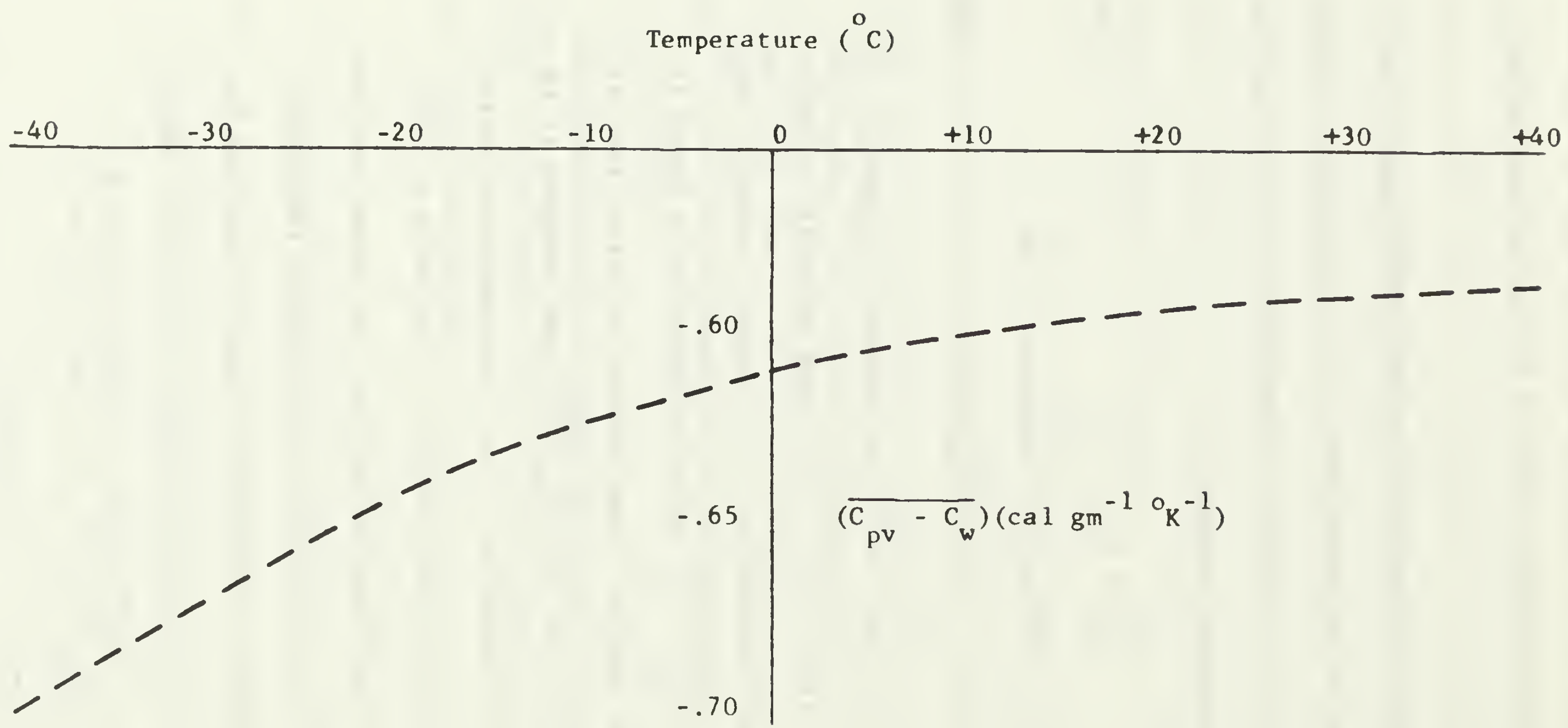

Figure 2: Variation of $(\overline{C_{pv} - C_w})$ with temperature.

Simplifying yields

$$\ln e_s = -\frac{A'}{T} - B' \ln T + C'$$

where

$$A' = \frac{\varepsilon\,[L_{vo} - T_o(\overline{C_{pv} - C_w})]}{R} = 6958.9262$$

$$B' = -\frac{\varepsilon\,(\overline{C_{pv} - C_w})}{R} = 5.65567$$

and

$$C' = \frac{\varepsilon}{R}\left[\frac{L_{vo}}{T_o} - (\overline{C_{pv} - C_w})\,(1+\ln T_o)\right] + \ln e_{so} = 59.01383$$

when $T_o = 233.16K$, $(\overline{C_{pv} - C_w}) = -0.62340$ cal $gm^{-1}\,{}^oK^{-1}$

and $L_{vo} = 621.7$ cal gm^{-1}.

Similarly, when freezing occurs one obtains

$$\ln e_s = \frac{-A''}{T} + C'' \qquad (2\text{-}7)$$

where

$$A'' = \frac{\varepsilon L_s}{R} = 6151.0205$$

and $C'' = \frac{A''}{T_o} + \ln e_{so} = 24.3277$ when $T_o = 233.16K$ and $L_s = 678$ cal gm^{-1}.

The above equations yield the saturation vapor pressure in millibars. Obviously the number of significant digits shown above exceeds the accuracy of some of the assumptions made to arrive at the values shown. Rounding to five significant figures could be done with no loss of accuracy, however they have been retained here as they were computed on the IBM OS/360 computer.

There were several minor changes made to the W.D. model. These included changing the value being used for the acceleration of gravity from 9.87 m sec^{-2} to a more exact form obtained by computing the value from the formula (List, 1951) expressing g as a function of latitude, altering the values used for the constants for conversion from degrees

Celsius to degrees Kelvin, conversion from meters to feet, and the value of the gas constant for dry air. In addition, the section of the program dealing with corrections due to the wind shear were completely eliminated which consequently allowed a subroutine (subroutine CHEQ) to be eliminated. Other minor changes included removing factors of 10 so that all pressure values are carried in units of millibars, introducing symbols wherever possible to simplify future modification and testing, changing the estimate of the updraft velocity at cloud base from 2.0 to 0.5 m sec^{-1}, and modifying certain format statements to improve the headings on the output. Finally, many soundings were run through the final modified version of the W.D. model to ensure that all changes had been properly programmed and to ensure that the results from the model were reasonable.

3. MATHEMATICAL DEVELOPMENT OF A FORECAST MODEL

This section describes the model used for this research. Although it is not consistent in many respects, it did provide a means by which forecasts of height at 1.5-hour intervals could be obtained for input to the W.D. model as well as provide a method by which the values of the various terms in the prediction equation could be ascertained. A description at the end of this section describes an attempt to set up the prediction equation in a consistent manner after the original model was running properly, however, the more consistent method failed to provide reasonable forecasts.

A. THE VORTICITY EQUATION

Following the procedures given by Haltiner and Martin (1957), the isobaric vorticity equation may be written as

$$\frac{d}{dt}(\zeta + f) = -(\zeta + f)\nabla\cdot\mathbb{V} + \left(\frac{\partial\omega}{\partial y}\frac{\partial u}{\partial p} - \frac{\partial\omega}{\partial x}\frac{\partial v}{\partial p}\right). \tag{3-1a}$$

Since the entire grid used for this research covers only 2 degrees of latitude, the Coriolis parameter can be considered constant, taking on the value it would have at the mean latitude of the grid. Using this approximation and expanding the left side of (3-1a) yields

$$\frac{\partial\zeta}{\partial t} = -\left(\omega\frac{\partial\zeta}{\partial p} + \mathbb{V}\cdot\nabla\zeta + (\zeta + f_m)\nabla\cdot\mathbb{V} + \frac{\partial\omega}{\partial x}\frac{\partial v}{\partial p} - \frac{\partial\omega}{\partial y}\frac{\partial u}{\partial p}\right). \tag{3-1b}$$

In order to use the W.D. model at 1.5-hour intervals, height of the base of the clouds is needed from which computations can be started. To obtain these heights, (3-1b) will be used as a height forecasting equation analogous to methods applied to large scale motions in the atmosphere by applying the balance equation. An alternative method without using the balance equation for initialization was tried and is discussed at the end of this section.

Since the application of (3-1b) to height forecasting will be done to obtain information regarding the orders of magnitude of the various terms and to obtain heights which can be used as cloud base information for input to the W.D. model, no attempt was made to set up a consistent set of equations. To obtain an equation which could be used for height forecasting and evaluation of the various terms with mesoscale data, the relative vorticity on the left side of (3-1b) and in the divergence term was replaced by $\nabla^2\psi$. In addition, only the rotational wind components were used for the horizontal advection term, that is,

$$u = -\frac{\partial\psi}{\partial y} \text{ and } v = \frac{\partial\psi}{\partial x}. \tag{3-2}$$

This was done because the present version has no provision for predicting the divergent component of the wind ($\mathbb{V}_\chi$). Future modifications should provide the means by which $\mathbb{V}_\chi$ can be used in addition to forecasts of the rotational component ($\mathbb{V}_\psi$). The vorticity in the vertical advection term as well as divergence and the u and v components of the wind in the twisting term were computed based on the observed winds at the initial time and held constant for 1.5 hours. Keeping values constant in the $\frac{\partial}{\partial p}$ terms greatly simplifies programming considerations since 3-minute time steps were used and to update these values at each time step would involve working with each level for only one time step before proceding to the next level. Unfortunately, holding ζ in the vertical advection term and the u and v components of the wind in the twisting term constant for 1.5 hours causes the seven levels to lack vertical coupling. Future modifications should attempt to correct this deficiency. After making these substitutions and assumptions, the resulting prediction equation becomes

$$\frac{\partial}{\partial t}(\nabla^2\psi) = -\left(\omega\frac{\partial\zeta}{\partial p} - \frac{\partial\psi}{\partial y}\frac{\partial\zeta}{\partial x} + \frac{\partial\psi}{\partial x}\frac{\partial\zeta}{\partial y} + (\nabla^2\psi + f_m)\nabla\cdot\mathbb{V} + \frac{\partial\omega}{\partial x}\frac{\partial v}{\partial p} - \frac{\partial\omega}{\partial y}\frac{\partial u}{\partial p}\right). \tag{3-3}$$

Assuming continuity in ψ, the term on the left side of (3-3) may be written as $\nabla^2 \frac{\partial\psi}{\partial t}$, and the equation then becomes a Poisson-type equation which can be solved by relaxation procedures as described by Haltiner and Martin (1957). The forecasting procedure, being one of successive iterations, will thus allow all terms involving ψ in (3-3) to vary at each time step.

To obtain a stream field for (3-3), consider the balance equation (Charney, 1955):

$$\nabla^2\psi + \frac{1}{f}(\nabla\psi\cdot\nabla f) - \frac{2}{f}\left[\left(\frac{\partial^2\psi}{\partial x\partial y}\right)^2 - \frac{\partial^2\psi}{\partial x^2}\frac{\partial^2\psi}{\partial y^2}\right] = \frac{\nabla^2\Phi}{f}. \tag{3-4}$$

Since the Coriolis parameter is to be considered constant over the grid, the second term of (3-4) drops out.

As discussed by Charney (1955) and Bolin (1955), equation (3-4) will maintain its elliptic character only if

$$\frac{\nabla^2\Phi}{f} + \frac{f}{2} > 0, \tag{3-5}$$

that is, only if the geostrophic relative vorticity is greater than $-f/2$. As shown in Appendix B, heights would have to be read at least to the nearest centimeter on this size grid in order to satisfy the requirements of (3-5) and still allow negative geostrophic relative vorticities. Consequently, (3-4) was forced to be elliptic by writing the non-linear terms in the form

$$\left(\frac{\partial^2\psi}{\partial x\partial y}\right)^2 = -\frac{\partial u}{\partial x}\frac{\partial v}{\partial y}, \tag{3-6a}$$

and

$$\frac{\partial^2\psi}{\partial x^2}\frac{\partial^2\psi}{\partial y^2} = -\frac{\partial v}{\partial x}\frac{\partial u}{\partial y}. \tag{3-6b}$$

Making this substitution into (3-4) and taking into account that the Coriolis parameter is to be considered a constant, (3-4) becomes

$$\nabla^2\psi = \frac{g}{f_m}\nabla^2 z - \frac{2}{f_m}J(u, v), \tag{3-7}$$

where J represents the Jacobian operator, and u and v are the components of the observed winds. Equation (3-7) is then an elliptic equation as it stands and does not require that the geostrophic relative vorticity be greater than -f/2. The ψ field can be initialized by the approximation $\psi = \frac{gz}{f_m}$, and then equation (3-7) can be relaxed to convergence before using (3-3) to predict the ψ fields at a later time. Similarly, once the ψ fields have been forecast ahead, the height fields can be recovered by making the approximation $z = \frac{f_m \psi}{g}$ and then utilizing (3-7) in the relaxation procedure until convergence is attained.

Convergence was defined as occurring when (3-7) balanced within 0.5 meters. When converting from height fields to stream fields this requires an epsilon of approximately 58612 $m^2\ sec^{-1}$, whereas epsilon becomes 0.5 m when converting from stream fields to height fields. These values arise from the approximation $\psi = (g/f_m)z$. In retrospect, the above convergence criteria may be too large for this scale of motion. In the future, more stringent criteria (such as 0.1 m) should be invoked.

As mentioned at the beginning of the section, attempts were made to correct the deficiencies of the prediction equation. This involved utilizing the exact relation $\nabla^2\psi = \zeta$ as the method of obtaining the initial ψ field instead of using the balance equation and to substitute $\nabla^2\psi$ for ζ in the horizontal advection term. In addition, the convergence criteria was changed to 10000 $m^2\ sec^{-1}$, in units of ψ, when converting from ζ to ψ.

The balance equation was used to recover height fields from the forecast fields of ψ with convergence criteria of 0.1 m. Two methods of initializing the ψ field were tried; they were $\psi = \frac{g\bar{z}}{f_m}$ and $\psi = \frac{gz}{f_m}$,

where $\bar{z}$ represents the standard atmosphere value of the height of the respective isobaric surfaces. When using $\bar{z}$ for the initialization, computations were carried out without difficulty, however the resulting forecast height fields were very flat as might have been anticipated. However, the forecast fields were in error by as much as 300 m at some levels. When using z for initialization, instability resulted in the horizontal advection term which caused an exponential growth of this term to values too large for the computer to handle after only 12-15 time steps (36-45 minutes ahead in time). The $\nabla^2\psi$ part of the horizontal advection term is believed to be the cause of the instability but this was not investigated in detail. However, one attempt to investigate the instability was made by shortening the time step to one minute. This delayed the instability somewhat which would indicate the problems were caused by computational instability.

B. VERTICAL MOTION COMPUTATIONS

In order to solve (3-3), vertical motion values are required for the vertical advection of relative vorticity and for the twisting term. Vertical motions for each of the seven levels were computed by means of the continuity equation in the form

$$\frac{\partial \omega}{\partial p} = - \overline{\nabla \cdot \mathbb{V}} \tag{3-8}$$

where the bar symbol denotes a vertical average through the layer. This method of computation utilizes the measured wind fields and does not allow the ω values to be updated for forecast times since the present model has no provisions for obtaining the divergent component of the wind. The vertical boundary conditions imposed were

$$\omega_{1000} = 0.0 \tag{3-9a}$$

and

$$\omega_{150} = 0.0 \tag{3-9b}$$

which also imply that $\nabla\cdot \mathbb{V}_{1000} = -\sum_{p=950}^{150} \nabla\cdot \mathbb{V}$.

Vertical motion computations were made by working downward from 150 mb through 300 mb and upward from 1000 mb through 700 mb. Vertical motion at 500 mb was then computed as the average of the values at 300 mb and 700 mb. It is recognized that using a backward or forward finite difference form for $\frac{\partial\omega}{\partial p}$ could at times lead to computational difficulties, however the method used here does allow the vertical motion at a given level to be affected by not only the divergence at that level and the level above, but also takes into consideration the divergence field at the level below.

Computations for the various levels were as follows (symbols such as W150 and W950 were chosen to conform to the symbols used in the computer program (subroutine VERTMO) shown in Appendix C):

(1) 150 mb

W150=0.0

(2) 200 mb ($\Delta p>0$)

$W175=W150-\Delta p\overline{\nabla\cdot \mathbb{V}}_{200-150}$

$W225=W150-\Delta p\overline{\nabla\cdot \mathbb{V}}_{300-150}$

W200=(W175+W225)/2

(3) 300 mb ($\Delta p>0$)

$W250=W200-\Delta p\overline{\nabla\cdot \mathbb{V}}_{300-200}$

$W350=W200-\Delta p\overline{\nabla\cdot \mathbb{V}}_{500-200}$

W300=(W250+W350)/2

(4) 1000 mb

W1000=0.0

(5) 950 mb ($\Delta p<0$)

$$W925=W1000-\Delta p\overline{\nabla\cdot \mathbb{V}}_{850-1000}$$

$$W975=W1000-\Delta p\overline{\nabla\cdot \mathbb{V}}_{950-1000}$$

$$W950=(W925+W975)/2$$

(6) 850 mb ($\Delta p<0$)

$$W825=W950-\Delta p\overline{\nabla\cdot \mathbb{V}}_{700-950}$$

$$W900=W950-\Delta p\overline{\nabla\cdot \mathbb{V}}_{850-950}$$

$$W850=(2(W825)+W900)/3$$

(7) 700 mb ($\Delta p<0$)

$$W675=W850-\Delta p\overline{\nabla\cdot \mathbb{V}}_{500-850}$$

$$W725=W850-\Delta p\overline{\nabla\cdot \mathbb{V}}_{500-950}$$

$$W700=(W675+W725)/2$$

(8) 500 mb

$$W500=(W300+W700)/2$$

The values obtained for vertical motion by this method are discussed in Section 5.

C. TEMPERATURE AND RELATIVE HUMIDITY FORECAST SCHEME

In order to utilize the W.D. model as an aid in severe storm forecasting, forecasted values of temperature and relative humidity are necessary in sounding form as input to the model.

Although a model designed for operational use should definitely be internally consistent, such as the vertical consistency of temperature fields with the geopotential fields, this study involved determining some of the problems that would develop when applying numerical forecasting techniques to mesoscale data. Consequently, the scheme for obtaining temperature forecasts which follows is not dynamically consistent with what has preceded, but it does provide a method of

obtaining a temperature-field forecast which could be used as input to the W.D. model. As discussed in Section 6, future studies should improve on this method.

Four different methods of forecasting temperature were tested and the results were then compared to verifying data. The method which verified best was selected as the forecast method for temperature, which was simply

$$\frac{\partial T}{\partial t} = - 1/2\, \mathbb{V}\cdot\nabla T, \qquad (3\text{-}10)$$

where the values for the wind components were taken as the measured winds at observation time and were not altered during the forecast interval.

Similar testing of four prediction methods was also done for relative humidity. The results led to

$$\frac{\partial R_h}{\partial t} = - 1/2\, \mathbb{V}\cdot\nabla R_h \qquad (3\text{-}11)$$

as the prediction equation for relative humidity where the wind components were again taken as the measured winds at observation time and were held constant for the forecast interval.

Temperature and relative humidity prediction calculations by equations (3-10) and (3-11) were performed in subroutine PROG, a listing of which may be found in Appendix C. Results of the computations, when compared to verifying data, are found in Section 5.

D. BOUNDARY CONDITIONS

To initialize the ψ field, the approximation $\psi \approx g\bar{z}/f_m$ was used as discussed in Section 3A. When the relaxation was performed, the outside row or column of the grid could not be modified due to finite

differencing. As a result, the boundary values of ψ were held at the value obtained by the initialization. The prognostic equation (3-3) was utilized from x = 3, 22 and y = 3, 22. After initializing the $\frac{\partial\psi}{\partial t}$ field to zero, it was then relaxed for the first time step. This relaxation was performed from x = 3, 22 and y = 3, 22 as mentioned above and the outer two rows or columns were maintained at their original values of zero. By this means, actual forecast values of ψ were obtained only from x = 3, 22 and y = 3, 22. After the first time step, the previous relaxed values for $\frac{\partial\psi}{\partial t}$ were used as the initial estimate for the new time step.

After obtaining the 1.5-hour forecast for the ψ field, the conversion back to z fields by means of (3-7) was accomplished by making the initial approximation $z \approx f_m \psi/g$ only for the points where the ψ field had been forecast. This then leaves the outer two rows or columns without forecasted height values. Before relaxing the z field, however, the values at these boundary points were set by performing a linear extrapolation from the fourth row or column outward as follows:

$$
\begin{array}{ll}
4 \;. & \\
3 \;. & \\
2 \;. & z_2 = z_3 - (z_4 - z_3) \\
1 \;. & z_1 = z_2 - (z_3 - z_2)
\end{array}
$$

Once this was accomplished at all of the grid points in the two boundary rows or columns, the z field was relaxed by means of the balance equation from x = 2, 23 and y = 2, 23.

Horizontal boundary values were not a problem with the vertical motion fields since they were computed from x = 2, 23 and y = 2, 23.

As mentioned above, the prognostic equation was utilized from x = 3, 22 and y = 3, 22 so that there were computed values of omega at x = 2, x = 23, y = 2 and y = 23 to use for the terms $\frac{\partial \omega}{\partial x}$ and $\frac{\partial \omega}{\partial y}$. As stated in (3-9), vertical boundary conditions for ω were that the vertical motion would vanish at 150 mb and 1000 mb, which then implies that the divergence at 1000 mb is equal to the negative of the sum of the divergences from 950 mb to 150 mb.

As will be discussed in Section 4, the initial data fields of height, temperature, relative humidity, wind direction and wind speed were analyzed on the computer by an objective scheme. The analysis method included finite differencing methods such that the outer row or column was not analyzed. These boundary values were set to the same value as the next interior row or column.

The same linear extrapolation technique that was used for the height fields was also used for the boundary conditions of the forecasted values of temperature and relative humidity, but it was only necessary to extrapolate from the second row or column out to the first row or column.

E. TIME STEPS

In the forecasting of the new ψ fields by (3-3), a 3-minute time step was used, thus requiring 30 time steps to obtain a 1.5-hour forecast. The method used was the leap-frog scheme described by Haltiner (upbl.), with a forward time step used for the first step. To eliminate the possibility of obtaining two separate solutions by the leap-frog method, the two solutions were averaged every 10 time steps

(30 minutes) and then one pass through the averaged field was made utilizing a Laplacian smoother with a .04 coefficient for smoothing. The leap-frog forecast method was then continued.

The forecasting of temperature and relative humidity by the simple advection schemes described in Section 3C does not involve any iterative type computations since the wind components were held constant for the forecast interval. Consequently, a single 90-minute forward time step was used. However, future modifications, which should include provisions for obtaining rotational and divergent components of the wind, should also update the wind field with each time step so that forecast values of the wind field could be used for temperature and relative humidity advection.

4. INITIALIZATION OF THE MODEL

In order to commence computations with the previously discussed model, 35 fields of data are needed (7 levels each of height, temperature, relative humidity, wind direction and wind speed). Such a volume of data would be difficult to obtain by manual means for real-time operational use of a model. Therefore, a more realistic approach is to use an objective analysis scheme by the computer.

Although many complicated and very sophisticated analysis schemes can be and have been developed (for example see Hughes, 1967), this was not the purpose of the research reported in this paper. Consequently, a simple objective analysis scheme was devised which uses the Laplacian operator and an eight-point averaging technique in alternating steps. Comparisons of computer analyses and hand analyses showed the patterns to be very similar. The major difference between the two analysis methods was the gradient in the vicinity of closed centers. The hand analyses tended to subjectively spread the gradient out over a much larger area whereas this was not so with the computer analyses. Which analysis is more nearly correct is merely a matter of subjective judgement. With only nine data points within a 14,400 square mile area during conditions of severe storms, this question will probably not be answered until the data network becomes more detailed. Nevertheless, the computer analyses were judged to yield adequate data fields for testing of the computer program.

In order to start the analysis a first guess was needed at each grid point. This was accomplished by dividing the grid into nine "regions of influence" (see Figure 3), each region containing one

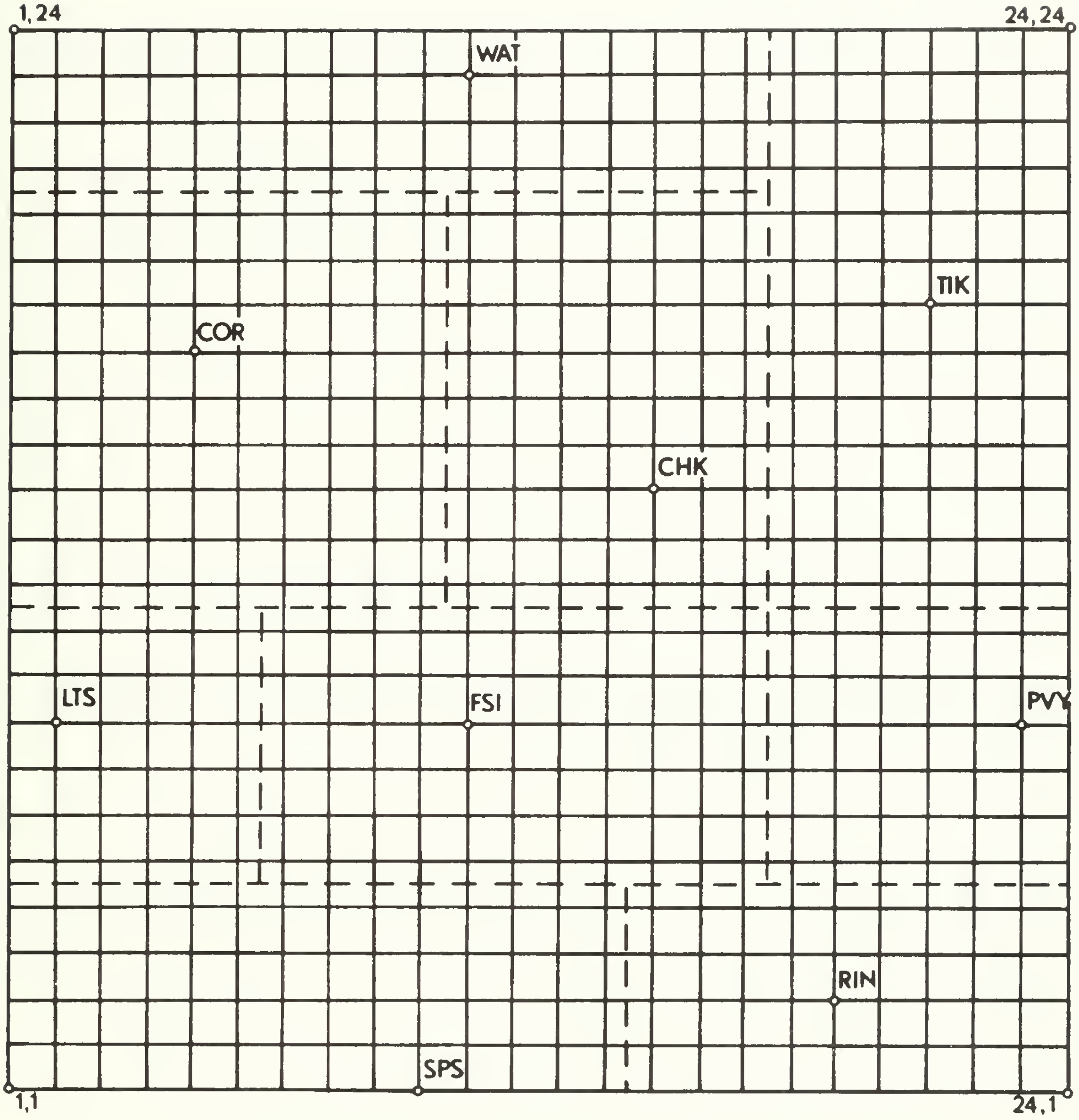

Figure 3: Regions of Influence.

of the nine radiosonde stations. The value of a given parameter at each grid point within a region was then set equal to the known value at the radiosonde station.

With the initial guess completed, the analysis was started. The first step was to alter the value at the grid points (except for the nine data points) by setting the value at that point equal to the average of the surrounding eight points according to the following diagram:

$$\begin{array}{ccc} \overset{4}{\cdot} & \overset{3}{\cdot} & \overset{2}{\cdot} \\ \overset{5}{\cdot} & \overset{0}{\cdot} & \overset{1}{\cdot} \\ \overset{6}{\cdot} & \overset{7}{\cdot} & \overset{8}{\cdot} \end{array}$$

$$P_0 = \frac{P_1 + P_2 + P_3 + P_4 + P_5 + P_6 + P_7 + P_8}{8} \qquad (4\text{-}1)$$

After making one pass through the grid in this manner, the Laplacian-type scheme was used such that

$$P_0 = P_0 + \delta\ (P_1 + P_3 + P_5 + P_7 - 4P_0), \qquad (4\text{-}2)$$

where the value of $\delta = 0.1$ was arbitrarily selected. After making ten passes through the grid in this manner, the eight-point averaging technique was again employed. The entire procedure was repeated in such a manner that the eight-point average was employed 11 times while the Laplacian scheme was utilized 100 times. The resulting analysis, after boundary conditions were set, was the analysis used in the prediction model. The boundary condition imposed was that the value at a grid point on an outside row or column was equal to the value at the adjacent grid point on the next row or column toward the center of the grid.

Figures 4a and 5a show hand analyses for two different levels and Figures 4b and 5b show the corresponding computer objective analyses. A copy of the program for the objective analysis scheme is included in Appendix C.

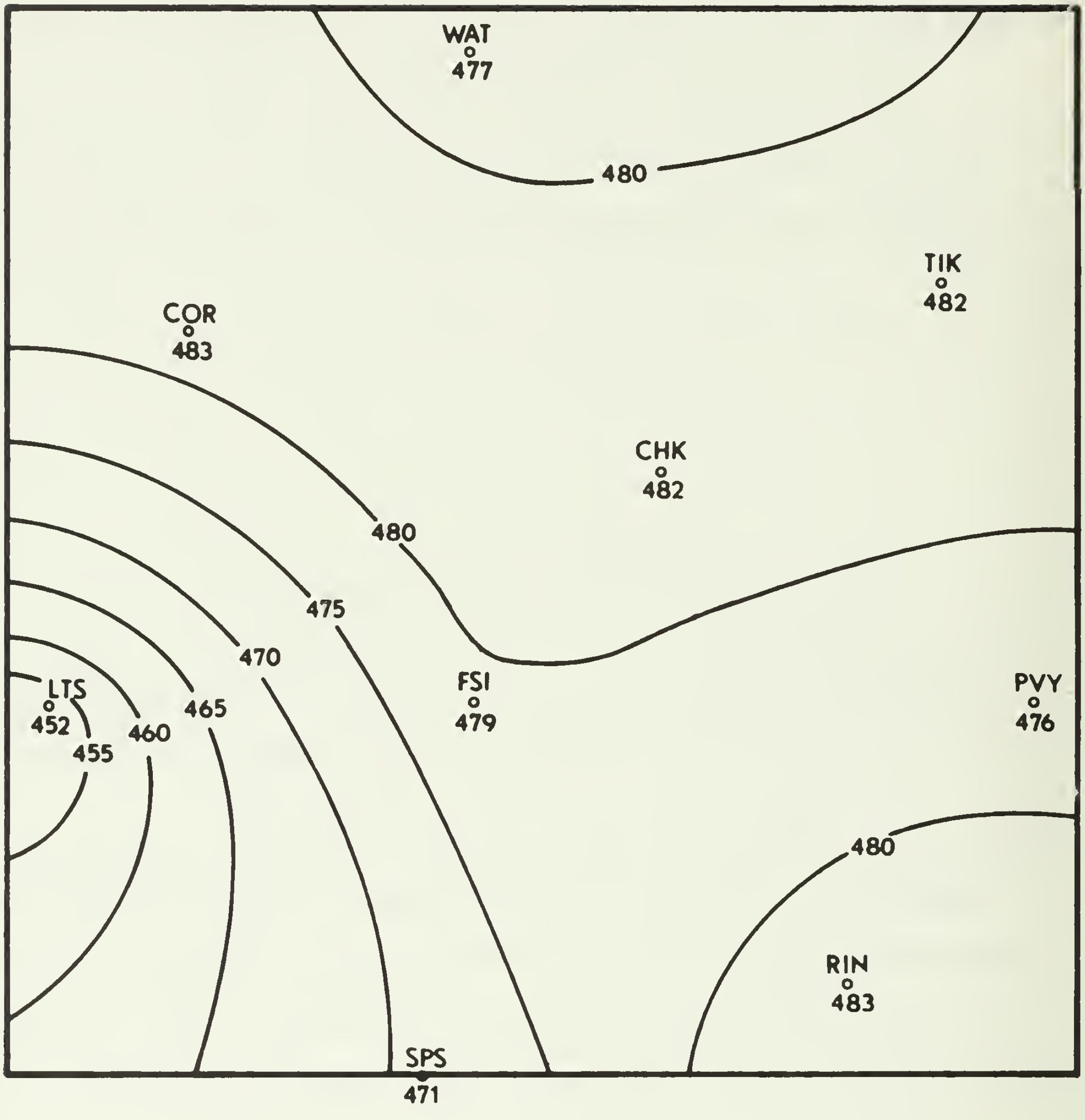

Figure 4a: Hand analysis for 950 mb at 1700Z on 30 May 1967. Contours every 5 meters.

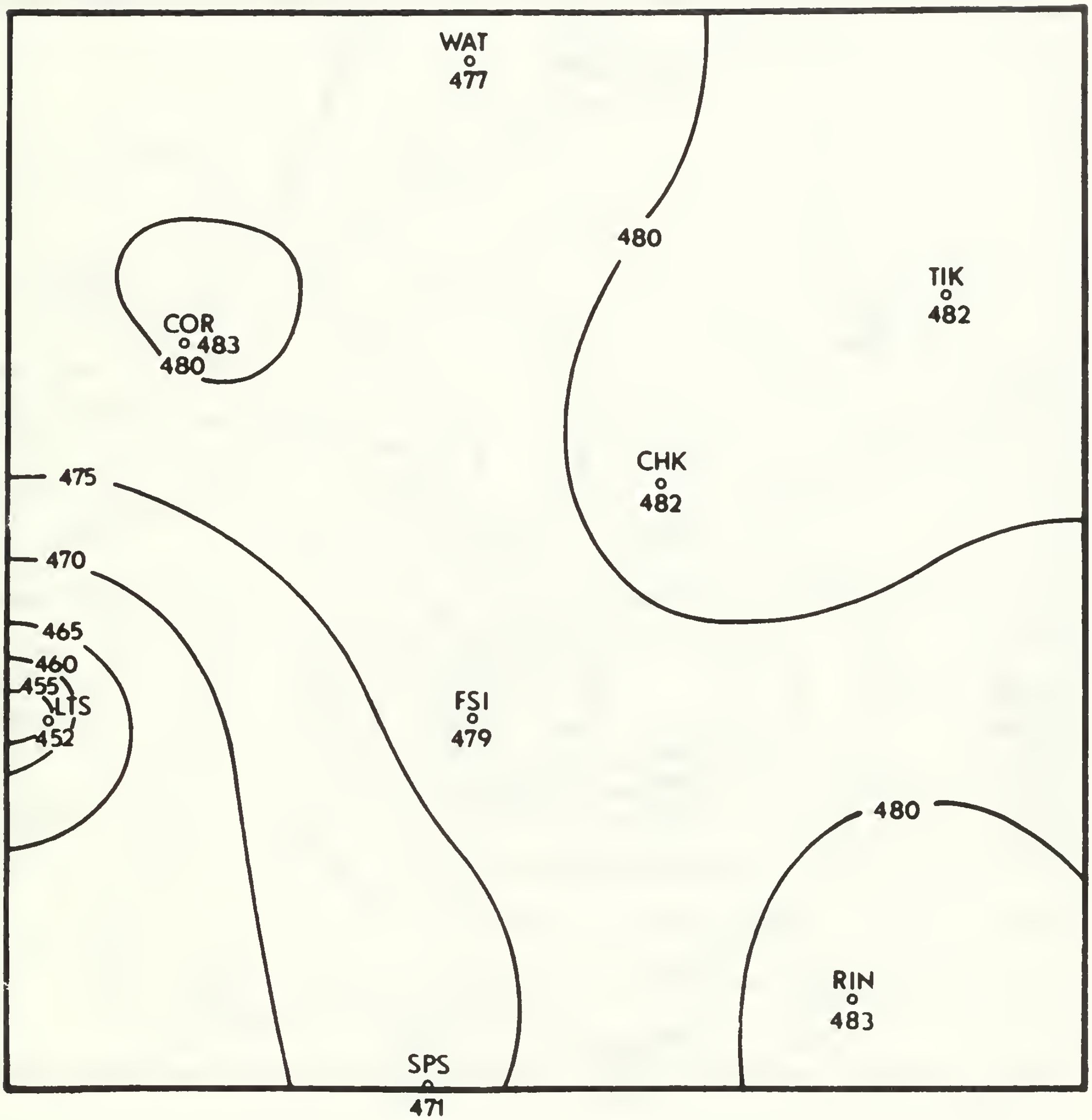

Figure 4b: Computer objective analysis for 950 mb at 1700Z on 30 May 1967. Contours every 5 meters.

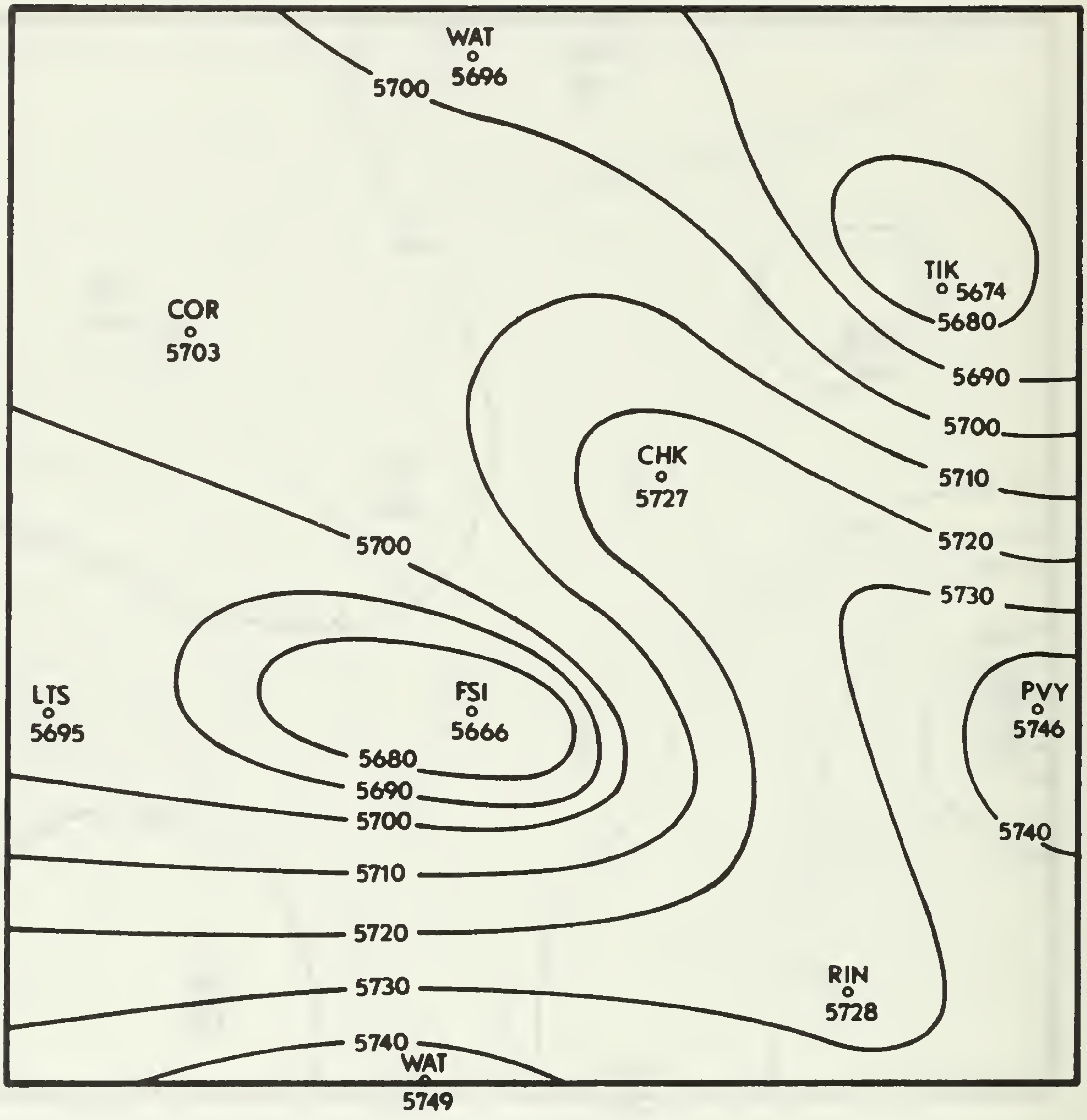

Figure 5a: Hand analysis for 500 mb at 1700Z on 30 May 1967. Contours every 10 meters.

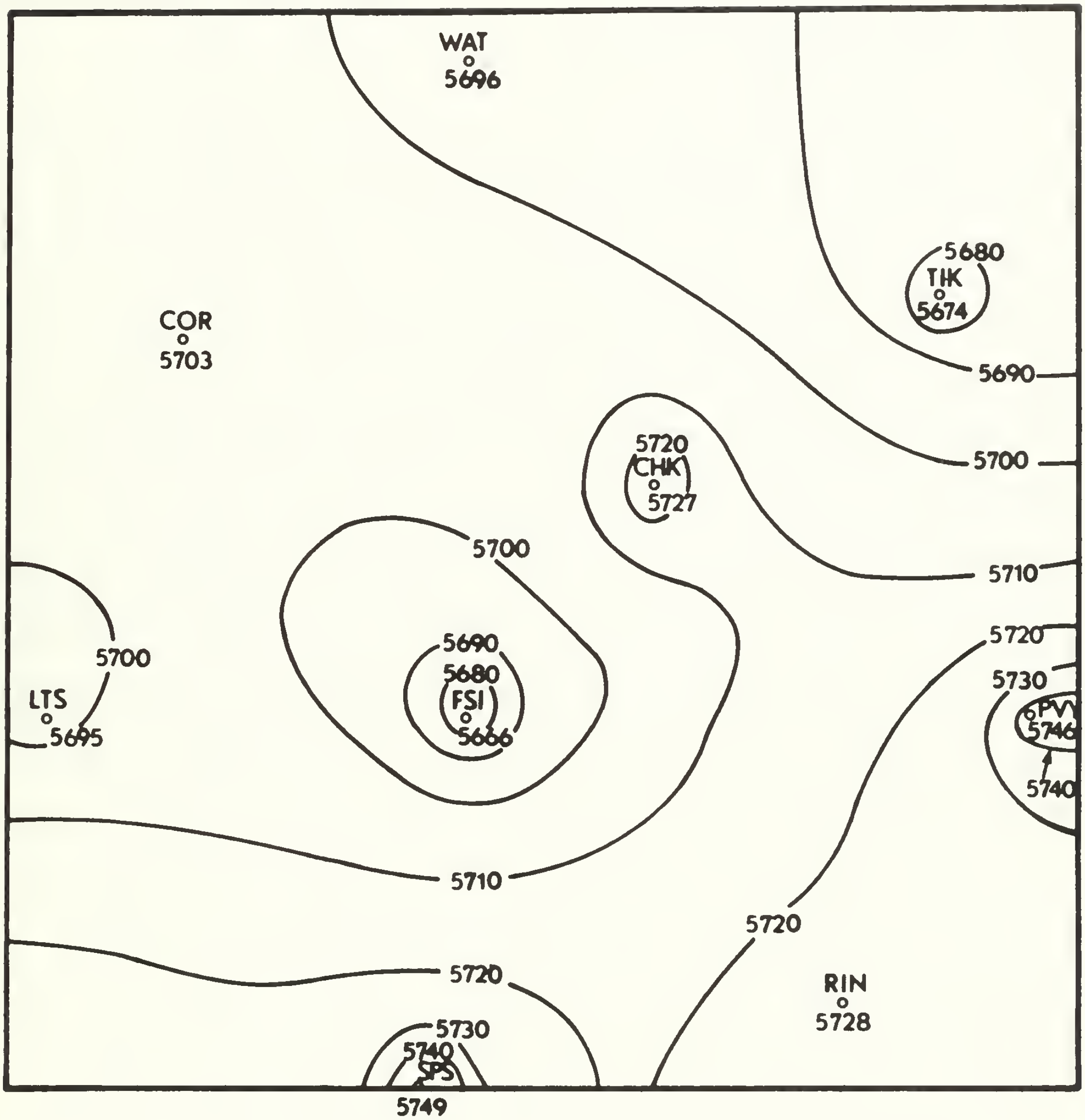

Figure 5b: Computer objective analysis for 500 mb at 1700Z on 30 May 1967. Contours every 10 meters.

5. RESULTS OF THE COMPUTATIONS

Two sets of data have been utilized with the previously described model. The times of the two sets of data are 1700Z, 30 May 1967 and 2130Z, 30 May 1967.

In synoptic-scale models, the terms in equation (3-1a) involving vertical motion and divergence have often been neglected since they may be considered relatively unimportant by scale analysis (Haltiner, upbl.) in the large scale motions of the atmosphere. However, when dealing with cumulus-scale data, this may not be true. In order to evaluate the relative importance of the various terms in the model throughout the entire grid, computations of the orders of magnitude of the various individual terms of equation (3-3) have been performed for both the 2130Z and 1700Z sets of observed data for the seven levels in the vertical described in Section 1. Tables 2 and 3 show the minimum and maximum values of each term based on the observed data, not on the forecast values. Although these tables only show the minimum and maximum values, they do point out the fact that the divergence term is very important. There were many instances when all five terms were of the same order of magnitude, that is, when each term was just as important as each of the other terms. Table 4 shows a sample of the values of each of the five terms as extracted from the computer printouts.

Out of 4800 values tested, $\frac{\partial\omega}{\partial x}\frac{\partial v}{\partial p}$ and $-\frac{\partial\omega}{\partial y}\frac{\partial u}{\partial p}$ were of the same sign 2584 times, or about 54% of the time. Therefore, in the cases tested, they were additive more often than not which is further justification that they should be retained for small-scale data.

PRESSURE LEVEL

TERM (UNITS OF 10^{-7})	950		850		700		500		300		200		150	
	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX
$\omega \frac{\partial \zeta}{\partial p}$	-.624	.206	-.258	.270	-1.62	.731	-.259	.286	-.192	.435	-.137	.067	0.0	0.0
$-\frac{\partial \psi}{\partial y}\frac{\partial \zeta}{\partial x} + \frac{\partial \psi}{\partial x}\frac{\partial \zeta}{\partial y}$	-6.04	9.81	-11.4	10.9	-24.3	20.9	-17.9	22.2	-13.8	19.2	-22.2	46.6	-8.16	12.7
$(\nabla^2 \psi + f_m)\ \nabla \cdot \mathbb{V}$	-11.7	11.9	-19.5	14.5	-47.2	55.0	-168.	74.4	-49.4	42.8	-147.	284.	-21.9	29.8
$\frac{\partial \omega}{\partial x}\frac{\partial v}{\partial p}$	-.369	.701	-.310	.195	-.909	1.41	-.539	.897	-.644	1.21	-.218	.066	0.0	0.0
$-\frac{\partial \omega}{\partial y}\frac{\partial u}{\partial p}$	-.321	.816	-.492	.351	-.933	1.50	-.246	.491	-1.26	.884	-.158	.163	0.0	0.0

Table 2: Minimum and maximum values of components of equation (3-3) for 1700Z data for all seven isobaric levels.

PRESSURE LEVEL

TERM (UNITS OF 10^{-7})	950		850		700		500		300		200		150	
	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX	MIN	MAX
$\omega \frac{\partial \zeta}{\partial p}$	-.532	.143	-.174	.074	-.203	.120	-.066	.108	-.130	.107	-.031	.068	0.0	0.0
$-\frac{\partial \psi}{\partial y}\frac{\partial \zeta}{\partial x} + \frac{\partial \psi}{\partial x}\frac{\partial \zeta}{\partial y}$	-5.82	8.70	-4.67	8.51	-13.4	10.9	-3.43	16.7	-12.7	22.0	-97.1	49.3	-47.2	24.5
$(\nabla^2\psi + f_m)\ \nabla \cdot \mathbb{V}$	-34.3	18.2	-22.0	23.6	-44.2	104.	-17.3	21.0	-47.3	164.	-200.	257.	-226.	68.9
$\frac{\partial \omega}{\partial x}\frac{\partial v}{\partial p}$	-.432	1.06	-.302	.992	-.356	.510	-.273	.347	-.564	1.47	-.160	.082	0.0	0.0
$-\frac{\partial \omega}{\partial y}\frac{\partial u}{\partial p}$	-.318	.503	-.366	.363	-.590	1.13	-.142	.100	-.438	.395	-.126	.105	0.0	0.0

Table 3: Minimum and maximum values of components of equation (3-3) for 2130Z data for all seven isobaric levels.

$\omega\, \partial\zeta/\partial p$	$-\frac{\partial\psi}{\partial y}\frac{\partial\zeta}{\partial x} + \frac{\partial\psi}{\partial x}\frac{\partial\zeta}{\partial y}$	$(\nabla^2\psi + f_m)\nabla\cdot\mathbb{V}$	$\frac{\partial\omega}{\partial x}\frac{\partial v}{\partial p}$	$-\frac{\partial\omega}{\partial y}\frac{\partial u}{\partial p}$
0.18E-08	-0.15E-08	-0.21E-09	-0.25E-08	0.22E-08
0.20E-08	0.24E-07	0.22E-09	-0.14E-08	0.64E-09
0.19E-08	0.56E-07	0.11E-08	-0.95E-09	-0.33E-09
0.11E-08	0.11E-06	0.32E-08	0.32E-09	0.91E-09
-0.30E-10	0.16E-06	0.88E-08	0.46E-08	0.36E-08
-0.20E-08	0.19E-06	0.14E-07	0.70E-08	0.21E-08
-0.43E-08	0,13E-06	0.20E-07	0.79E-08	-0.31E-09
-0.61E-08	0,34E-07	0.20E-07	0.66E-08	0.95E-10
-0.73E-08	-0,16E-08	0.15E-07	0.51E-08	-0.84E-09
-0.69E-08	-0,11E-07	0.68E-08	0.43E-08	-0.63E-09
-0.75E-08	-0,16E-07	0.24E-08	0.28E-08	-0.62E-10
-0.67E-08	0,99E-08	-0.45E-09	-0.35E-09	0.98E-11
-0.73E-08	0,28E-07	-0.14E-08	-0.19E-08	0.59E-09
-0.40E-08	0,27E-07	-0.19E-08	-0.16E-07	0.29E-08
-0.11E-08	0,23E-07	-0.28E-08	0.43E-08	0.45E-08
0.92E-09	0,11E-07	-0.34E-08	0.61E-08	0.39E-08
0.13E-08	0,50E-08	-0.34E-08	0.31E-08	0.12E-08
0.84E-09	0,18E-08	-0.30E-08	0.21E-09	-0.14E-08
0.30E-09	0.14E-08	-0.20E-08	-0.28E-09	-0.37E-08
0.41E-09	-0,41E-07	-0.12E-08	-0.82E-09	0.79E-08
0.13E-08	-0,10E-07	-0.13E-09	-0.12E-08	0.31E-08
0.10E-08	0,24E-07	0.14E-09	-0.23E-08	0.24E-09
0.56E-09	0,54E-07	0.59E-09	-0.30E-08	-0.22E-09
-0.25E-09	0,11E-06	0.26E-08	-0.26E-08	0.46E-08
-0.15E-08	0,22E-06	0.80E-08	0.26E-08	0.75E-08

Table 4: Sample of values of the components of equation (3-3) at 950 mb for 1700Z data.

Vertical motion computations yielded considerably different values for the 1700Z data and the 2130Z data. Table 5 shows the maximum upward and downward vertical motions for each isobaric level at each of the times. Although the values in Table 5 may, at first, seem somewhat large, a transformation by the formula

$$\omega \approx -\rho\, gw \qquad (5\text{-}1)$$

where ρ is replaced by p/RT from the equation of state to yield

$$\omega \approx -\frac{p}{RT}\, gw, \qquad (5\text{-}2)$$

shows that a value of ω = 200 mb hr^{-1} upward corresponds to a value of $w \approx .86$ m sec^{-1} upward (at 500 mb) which is certainly a reasonable value for cumulus convection. Although values this large were only present at a few points, further research is needed to determine what reasonable values are for this scale of motion. In general, the vertical motion fields showed fairly strong upward vertical motions throughout the grid at 1700Z and correlated fairly well with the squall-line that was present at that time. There were a few areas in which the correlation was not too good but this may possibly be due to the fact that the nine radiosondes were not all launched simultaneously and a difference of 15 minutes between launch times could possibly result in the bad correlation regions. Although the radar pictures indicated that dissipation of the squall-line had begun at 2130Z by the fact that isolated echoes were present rather than a large mass of clouds, the vertical motion fields did not show extensive regions of downward motions except at 300 mb. The 2130Z values were, however, considerably less in intensity than they were at 1700Z which might have been anticipated from the results shown in Table 5.

Pressure Level / Up/Down/Time	950	850	700	500	300	200
Max. Upward 1700Z	-98	-74	-220	-105	-108	-31
Max. Downward 1700Z	+78	+86	+219	+55	+76	+21
Max. Upward 2130Z	-48	-30	-110	-39	-53	-18
Max. Downward 2130Z	+52	+46	+90	+72	+78	+10

Table 5: Maximum upward and downward vertical motions for each isobaric level at 1700Z and 2130Z on 30 May 1967 in mb hr^{-1}.

The significance of the much larger values of vertical motion at 1700Z as compared to 2130Z was not investigated thoroughly, however larger values of divergence were evident (as would be expected) at 1700Z. This was particularly true at 200 mb, and this would tend to be propagated downward to 300 mb and 500 mb by the computation procedure.

The simple method of forecasting temperature resulted in values which verified slightly better than a persistence forecast. The average absolute error (not considering whether the error was an overestimate or an underestimate) for 97 persistence temperature forecasts was 1.8C while for 105 50% temperature advection forecasts the average absolute error was 1.4C. The reason for the different number of verifications is due to the fact that some of the 1700Z and 2130Z radiosonde data terminated before reaching 150 mb. Verifications were performed only at the nine radiosonde stations, and only at each of the seven discrete isobaric levels.

The simple method of forecasting relative humidity resulted in values which verified essentially the same as a persistence forecast. The average absolute error for 97 persistence relative humidity forecasts was 18.0% while for 105 50% relative humidity advection forecasts the average absolute error was 18.2%.

The results of the height forecasts did not verify as well as was expected. This is undoubtedly due in part to the fact that additional refinements are necessary in the prediction method to obtain a consistent set of equations. Table 6 compares the average absolute errors of persistence and model 1.5-hour forecasts for both the 1700Z and the 2130Z sets of data. As can be seen in the table, the model did perform slightly better than persistence 50% of the time.

AVERAGE ABSOLUTE ERROR (METERS)

TYPE OF FORECAST (VERIFYING TIME)	950	850	700	500	300	200	150
Persistence (1830Z)	4.5	8.4	18.2	26.0	26.8	31.4	34.2
Model (1830Z)	4.4	7.1	8.8	28.5	28.1	31.8	31.0
Persistence (2300Z)	13.1	8.9	10.6	6.5	31.1	44.5	53.1
Model (2300Z)	12.9	9.1	7.2	14.9	37.5	32.0	58.5

Table 6: Average absolute errors of height fields (meters) for persistence and model at 1700Z and 2130Z for all seven isobaric levels.

As previously mentioned, the W.D. model yields profiles of vertical velocity, hydrometeor water content and temperature excess. In addition, total rain expected from the cloud, duration of the precipitation, updraft area, and a profile of updraft radius are included in the results of the computations. Figure 6 shows a sample of the graphical output of the W.D. model. These results are printed out in graphical form by the two subroutines (GRPHCL and GRAPH) which were included with the W.D. model (Weinstein and Davis, 1967). The horizontal axis (shown in Figure 6 at 1/2 scale) shows the scale to be used for the various parameters and the vertical axis on the left shows the corresponding heights in meters, with the bottom figure (in this case 1642) being the first increment (200 meters) above cloud base. In the actual computer output of Figure 6, the lines are represented by letters, such that the vertical velocity curve is a series of the letter "W", hydrometeor water content of the letter "Q", temperature excess of the letter "T", updraft radius of the letter "R", and the zero axis for the temperature excess curve is represented by the symbol "$".

The W.D. model has a provision for setting a constant value for the updraft radius or for allowing it to vary within a cloud. However, for this research the updraft radius was always considered to be constant and equal to 1 km. The profile shown in Figure 6 was computed based upon the 1830Z, 30 May 1967 forecasted values of temperature, relative humidity and height of the cloud base, and by assuming that cloud base was at 850 mb for station CHK.

Many soundings were used as input to the W.D. model, and in many cases the values of the temperature excess determined by the model seemed to be too large. Values as high as 4 or 5 degrees Celsius

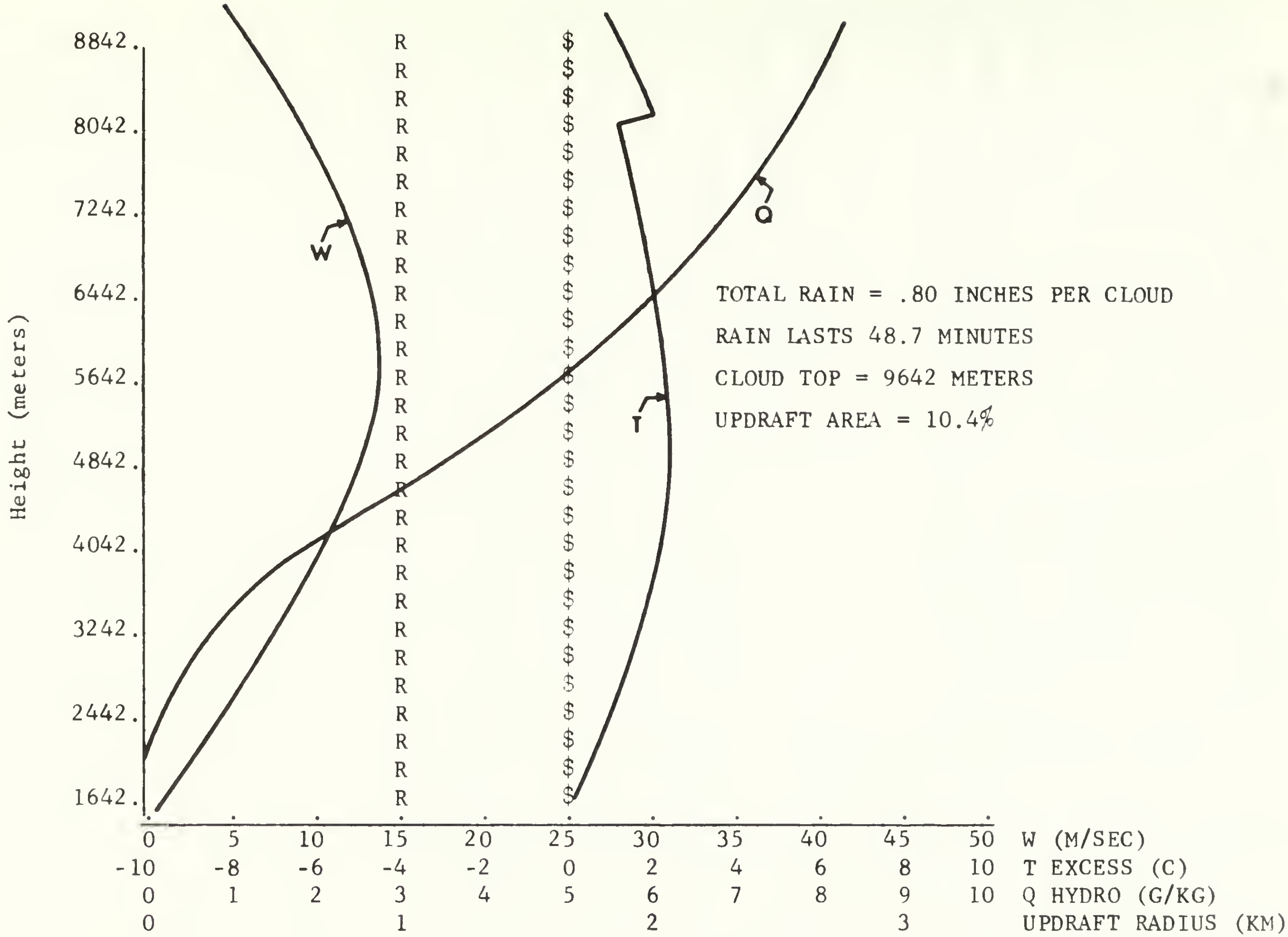

Figure 6: Sample of output from W.D. model (Weinstein and Davis, 1967).

occurred quite often, with an occasional value of 7 or 8 degrees Celsius. This intuitively seems to be too high but additional observations are needed to determine if such values actually exist within cumulus clouds. In addition, apparent discontinuities in temperature excess, as evidenced near 8000 meters on Figure 6, are quite common in the graphical output of the model. These large temperature excess values and temperature excess discontinuities occurred not only with forecast soundings but also with actual observed soundings from the NSSL data.

Figure 6 is based upon an ice nucleation temperature of -25C which is assumed to be representative of natural conditions in the atmosphere. The W.D. model allows for changing the ice nucleation temperature to a value of, say, -6C to represent the conditions that might occur if cloud seeding were to be undertaken. However, this is presently not included in the subroutine version of the W.D. model (subroutine CLOUDS) since the reason for using it as part of a prediction model was to aid in severe storm forecasting.

6. SUGGESTIONS FOR FUTURE DEVELOPMENTS AND IMPROVEMENTS

The first modification to the previously discussed model, which became evident during testing of the model, would be to expand the horizontal grid dimensions to at least 32 x 32. Figure 3 illustrates that four of the nine grid points lie within the first two rows of the boundaries. This more or less negates application of the prognostic equations to these grid points due to finite differencing. It is estimated that this improvement would increase the storage requirements of the program by about 50%, and since the program is already quite large, modifications would have to be made to allow for the sharing of some storage locations wherever possible.

At present the model lacks the proper provisions for predicting the complete wind field and for updating it after each time step. A consistent scheme for updating the wind field should be included such that the predicted winds could be recovered every time step. In addition, this would allow predictions further ahead in time as well as allow for updating the vertical motion fields with new divergence fields.

As discussed in Section 3A, the present version of the forecast model does not provide for vertical coupling of the seven isobaric levels since the vorticity in the vertical advection term and the u and v components of the wind in the twisting term were held constant for the 1.5-hour forecast interval. To correct this deficiency would involve performing one time step at a given level before proceding to the next level. Also, additional storage would be required to save the field which $\frac{\partial}{\partial p}$ operates on for one time step back so that vertical derivatives could be taken at each level on fields which all correspond

to the same time. As far as programming considerations are concerned, this essentially involves nesting the vertical level DO LOOP inside of the time step DO LOOP.

When the model has been modified to allow for vertical coupling of the seven isobaric levels, additional levels in the vertical could be included so that vertical layer depths would not exceed 100 mb. This would require perhaps 10 or 11 levels which would increase the storage requirements of the computer considerably, however, this would allow a much finer detail for the forecasted soundings. In addition, it should aid in the delineation of the vertical velocities at each level.

A consistent dynamical method should be included for the prediction of temperature and relative humidity. The present method is more empirical than dynamical and better verifications should be obtained if a more realistic approach is devised. As an alternative, a better constant (vice 50%) might be tested, particularly in the case of relative humidity where the errors were negative much more often than they were positive.

In subroutine SETUP (see Appendix C), cloud base is assumed to be at 850 mb for entry into subroutine CLOUDS (see Appendix C). A routine could be developed to be included in SETUP such that the lifting condensation level, mixing condensation level or convective condensation level would be computed as the cloud base before entering the W.D. model (subroutine CLOUDS). Comparison of results obtained for cloud base at 950 mb and 850 mb indicate that the cloud base and the associated temperature and relative humidity are very critical to the results obtained from the W.D. model.

Once a complete internally consistent model is set up, calculations out past 1.5 hours could be tried to determine at what point such a model loses its skill. A few runs were made out to 4.5 hours and the results appeared to indicate values of height which were too large, and the temperature fields appeared to become unstable.

Kessler (1967) proposed the formula for the fall speed of precipitation to be of the form

$$V = 5.0939\ M^{.125}. \tag{6-1}$$

The corresponding formula in the W.D. model is

$$V = 15.39\ Q_h^{.125}. \tag{6-2}$$

The units of M are gm m^{-3} whereas the units of Q_h are gm gm^{-1}. In order to convert M to the units of Q_h, it is necessary to divide by the density of air, ρ. Weinstein and Davis (1967) have apparently selected a representative value for ρ. It is recommended that future modifications of the W.D. model include the density in the computation. Since $M = \rho Q_h$, and $\rho = p/RT$ from the equation of state, it would be a simple change in the programming to use the following formula for the hydrometeor terminal velocity:

$$V = 5.0939 \left(\frac{pQ_h}{RT}\right)^{.125}. \tag{6-3}$$

Since the duration calculation is based on the hydrometeor terminal velocity at cloud top, (6-3) will permit values of V which not only depend on the hydrometeor liquid water content but also will depend on the pressure and temperature at the cloud top.

Finally, it has been suggested to the author by Dr. J. D. Mahlman (personal communication) that the only solution to overcome the many difficulties with this scale of motion would be a direct integration

of the primitive equations. It is believed that no such research has been attempted for mesoscale data and this approach definitely warrants consideration for future attempts at numerical modelling of such scales of motion.

7. SUMMARY

A numerical weather prediction model has been described for use with cumulus-scale data. Together with the W.D. model, it provides forecasts of height, temperature, and relative humidity at seven isobaric levels as well as vertical profiles of vertical velocity, temperature excess and hydrometeor liquid water content at 1.5-hour intervals. The model was tested on two sets of data and yielded forecasts which were, in general, statistically about the same as a persistence forecast. This can probably be attributed in part to the fact that the present version has no provisions for coupling the seven isobaric levels nor for updating the vertical motion fields after each time step. Another factor is that the temperature and relative humidity forecast schemes were more empirical than dynamical, and were not consistent with the vorticity equation. This may account for the lack of skill in forecasting these parameters. Suggestions were made to correct some of these inconsistencies in the event that further research is conducted with the model. However, the possibility of applying numerical forecasting techniques to small-scale data does appear to have possibi ities but extensive research is needed in this area before useful forecasts will be obtained.

The model requires approximately 5.5 minutes on the IBM OS/360 (MVT versions 15, 16) computer to yield a 1.5 hour forecast. An additional 14 seconds per field is required for the objective analysis scheme described in Section 4, thus requiring about 13.5 minutes for the analysis of 35 fields of data and the prediction of the aforementioned parameters for 1.5 hours.

REFERENCES

Barnes, S. L., 1968: On the source of thunderstorm rotation. Tech. Mem. ERLTM-NSSL 38, 30 pp.

Bolin, B., 1955: Numerical forecasting with the barotropic model. Tellus, 7, 27-49.

Charney, J., 1955: The use of the primitive equations of motion in numerical weather prediction. Tellus, 7, 22-26.

Haltiner, G. J., Numerical Weather Prediction, unpublished notes, The Naval Postgraduate School, Monterey.

Haltiner, G. J., and F. L. Martin, 1957: Dynamical and Physical Meteorology. McGraw-Hill Book Co., Inc., New York.

Hammond, G. R., 1967: Study of a left moving thunderstorm. Tech. Mem. IERTM-NSSL 31, 75 pp.

Hughes, R. E., 1967: U. S. Naval Weather Service Computer Products Manual, NAVAIR 50-1G-522.

Kessler, E., 1967: On the continuity of water substance. Tech. Mem. IERTM-NSSL 33, 125 pp.

List, R. J., Ed., 1951: Smithsonian Meteorological Tables. Smithsonian Institution.

Newton, C. W., and J. C. Fankhauser, 1964: Movement and development patterns of convective storms, and forecasting the probability of storm passage at a given location. NSSP Report No. 22, 53 pp.

Stommel, H., 1947: Entrainment of air into a cumulus cloud. Jour. Met., 4, 91-94.

Weinstein, A. I., and L. G. Davis, 1967: A parameterized numerical model of cumulus convection. Report No. 11 to the NSF, (NSF GA-777), Dept. of Met., The Penn. State Univ., 66 pp.

APPENDIX A

THE NSSL RADIOSONDE NETWORK

There were 10 radiosonde stations within the NSSL region considered for this study which are listed below. The grid-point numbers associated with each station are shown on Figure 3, where the stations are shown by their call letters.

Call letters	Name	Location Lat.	Location Long.
SPS	Sheppard AFB, Texas	33.97	98.48
RIN	Ringling, Oklahoma	34.17	97.58
LTS	Altus, Oklahoma	34.70	99.33
FSI	Fort Sill, Oklahoma	34.65	98.40
PVY	Pauls Valley, Oklahoma	34.70	97.22
COR	Cordell, Oklahoma	35.30	98.97
CHK	Chickasha, Oklahoma	35.10	97.97
OKC	Oklahoma City, Oklahoma	35.40	97.60
TIK	Tinker AFB, Oklahoma	35.42	97.38
WAT	Watonga, Oklahoma	35.85	98.42

All of these stations, except OKC, were used in this study. The reason for not including data from OKC is that their data corresponds to synoptic data times rather than at 1.5-hour intervals.

APPENDIX B

ON THE ELLIPTICITY REQUIREMENT OF THE BALANCE EQUATION

The ellipticity requirement of the balance equation (Charney, 1955 and Bolin, 1955) was given by equation (3-5) which is

$$\frac{\nabla^2 \Phi}{f} + \frac{f}{2} > 0. \tag{3-5}$$

Bolin (1955) stated that the data at grid points which did not satisfy (3-5) was modified so that (3-5) was satisfied. This section investigates the extent to which grid distance influences the requirement of (3-5).

Expanding (3-5) results in

$$\frac{m^2 g}{d^2 f_m} \nabla^2 z + \frac{f_m}{2} > 0 \tag{B-1}$$

where

∇^2 = finite difference form of the Laplacian operator,

m = map factor $\approx$ 1.19,

g = gravity $\approx$ 9.80,

d = grid distance,

z = height,

and f_m = mean Coriolis parameter = $.836 \times 10^{-4}$.

Upon substitution of the above values for each factor, (B-1) reduces to

$$\nabla^2 z + .00087 d^2 > 0 \tag{B-2}$$

where z is in meters and d is in nautical miles. Substituting a value of d = 5 nm yields

$$\nabla^2 z > -.02175. \tag{B-3}$$

By measuring heights to the nearest meter, the only way to satisfy (B-3) is to have $\nabla^2 z \geq 0$. To satisfy (B-3) on a grid with d = 5 nm

requires z to be measured at least to the nearest .01 m, and this would still not guarantee meeting the requirement. It is easily shown with equation (B-2) that the smallest grid distance on which (3-5) can be satisfied, without eliminating the possibility of $\nabla^2 z < 0$ and still measure z to the nearest meter, is 35 nm.

The result from above is what prompted the substitution of the measured wind field into the balance equation in order to force it to be elliptic. Tests were made to determine with what frequency the balance equation ellipticity requirement was satisfied with this scale of data. The results show that out of 6776 points, only 62.4% satisfied the requirement. At 1700Z the requirement was satisfied 59.4% of the time while at 2130Z the figure was 65.5%.

APPENDIX C

This appendix includes a listing of the computer program used for the research described by the preceding sections. In order to aid any possible research in the future with this program, each subroutine will be listed with a brief explanation of the calling arguments.

1. Subroutine VERTMO

   U = Zonal component of the horizontal wind in m sec^{-1}.

   V = Meridional component of the horizontal wind in m sec^{-1}.

   OMEGA = ω = Vertical component of the wind in (x, y, p, t) coordinate system in mb sec^{-1}.

   NI = Number of grid points along x-axis.

   NJ = Number of grid points along y-axis.

   NK = Number of levels in the vertical.

   DIST = Distance between grid points in meters.

   DSI = The negative of the divergence field for 1000 mb = the sum of the divergence from 950 mb to 150 mb.

2. Subroutine FORFUN

   VORT = Vertical component of relative vorticity.

   F = Field for forcing function.

   FMAP = Map factor (mean).

   FBAR = Coriolis parameter (mean).

   DIST = Distance between grid points in meters.

   DT = Forward time step in seconds.

   USPD = Zonal component of the horizontal wind in m sec^{-1}.

   VSPD = Meridional component of the horizontal wind in m sec^{-1}.

   K = The level the main part of the program is at when this subroutine is called.

IZ = Counter to determine at what time step (1-30) the main program is.

PSI = The ψ field in m^2 sec^{-1}.

OMEGA = ω = Vertical component of the wind in (x, y, p, t) coordinate system in mb sec^{-1}.

3. Subroutine RELAXI

M = Number from 2, NI = a number one greater than what is needed in DO LOOP.

N = Same as M except 2, NJ.

F = $\Delta\psi$ field for relaxation of balance equation.

A = ψ field in m^2 sec^{-1}.

B = Z field in meters.

L = Counter to keep track of number of points which have converged.

EPS = Epsilon = convergence criteria.

ALPHA = Over-relaxation constant.

FMAP = Map factor (mean).

FBAR = Coriolis parameter (mean).

DIST = Distance between grid points in meters.

IL = Lower point for DO LOOP in x direction.

JL = Lower point for DO LOOP in y direction.

LM = Iteration number from main program.

U = Zonal component of the horizontal wind in m sec^{-1}.

V = Meridional component of the horizontal wind in m sec^{-1}.

K = Level in the vertical from main program.

G = Gravity.

4. Subroutine RELAX

Same as in RELAXI except:

F = Forcing function.

R = Residual field.

ICOUNT = L.

I1 = IL.

J1 = JL.

5. Subroutine SMOOTH

D = Field to be smoothed.

RF = Smoothing factor (<1.0).

IPASS = Number of smoothing passes to make through the grid.

IL, JL = Lower point for DO LOOP.

IH, JH = Upper point for DO LOOP.

6. Subroutine PROG

TEMP = Temperature.

RH = Relative humidity.

DIR = Wind directions.

SPD = Wind speeds.

USPD = Zonal component of SPD in m sec^{-1}.

VSPD = Meridional component of SPD in m sec^{-1}.

DIST = Distance between grid points in meters.

NK = Number of levels in the vertical.

NI = Number of grid points on x-axis.

NJ = Number of grid points on y-axis.

NI1 = NI-1.

NJ1 = NJ-1.

OMEGA = ω.

Z = Height field.

7. Subroutine SETUP

TEMP = Temperature.

RH = Relative humidity.

Z = Height field.

IL = A number from 1 to NI, usually 1.

IH = A number from 1 to NI, usually NI.

JL = A number from 1 to NJ, usually 1.

JH = A number from 1 to NJ, usually NJ.

8. Subroutine CLOUDS.

This is the modified version of the W.D. model (Weinstein and Davis, 1967).

PDATA = Pressure levels.

ZDATA = Height field.

RHDATA = Relative humidity field.

ZZ1 = Height of cloud base.

DZ = Vertical increment (meters) to be used.

NCALL = Number of times this subroutine called. Used as variable to bypass a READ statement.

IPT = I coordinate for set of data coming in.

JPT = J coordinate for set of data coming in.

9. Calling arguments of GRPHCL and GRAPH are all parameters which have been computed or set in subroutine CLOUDS.

10. Subroutine METMAP

This subroutine was not written by the author, but is part of the Naval Postgraduate School Computer Facility Library and has been included for information. It is simply a shading routine that will print out a field of data with a 0.5 inch grid spacing and contour the field.

Y = Two-dimensional field to be contoured.

N = Number of rows I in the array to be contoured.

M = Number of columns J in the array to be contoured.

T = Title for printout . Up to 96 columns of alpha-numeric information.

BND = Bandwidth desired for contouring.

AZ = Scaling constant; that is, each value will be multiplied by AZ before printing it out. Only the first three numbers to the right of the decimal point are on the output, so AZ can be used to control which numbers are to be printed out.

BZ = Additive constant. This is useful when working with D-values. Usually, however, it is 0.0. If used, this constant will be added to each value of Y before printing it out.

AMIN = Minimum value for subroutine to start contouring.

IJT = 0 means subroutine will compute the minimum value and start contouring at that value. A value must still be specified for AMIN but it will not be used.

ICON = 1 if contouring desired.

= 0 if no contouring desired.

When working with this subroutine, it must be remembered that the grid point I = 1, J = 1, is the upper left corner of the grid, and I increases downward while J increases to the right. If an array is defined with the lower left corner as the point (1, 1), the following set of FORTRAN statements will get the array in the proper order for the subroutine:

```
      DO 5 I=1, 24
      DO 5 J=1, 24
    5 DUMMY (J,I)=Z(I,24-J+1)
```

The field called DUMMY is then taken into METMAP as Y.

```
C
C   THIS IS THE MAIN PROGRAM TO OBTAIN PROGS OF THE HEIGHT,TEMPERATURE,
C   AND RELATIVE HUMIDITY FIELDS FOR THE NSSL NETWORK IN OKLAHOMA NEAR
C   35 DEGREES LATITUDE. A 24X24 GRID IS USED CENTERED ABOUT 5 MILES NE
C   OF STATION 5C  ORIENTED 000/180 AND 090/270 DEGREES TRUE. ALL
C   PARAMETERS WHICH ARE A FUNCTION OF LATITUDE WILL BE TAKEN AS CONSTANT
C   THROUGHOUT THE GRID WITH THE VALUE AT 35 DEGREES.
C   DY= 5 NAUT. MILES AND DX= 6 MINUTES OF LONGITUDE=5 MILES (APPROX).
C   I=1, J=1 IS LOWER LEFT CORNER OF GRID.  I INCREASES TOWARDS THE
C   RIGHT AND J INCREASES MOVING UPWARDS
C
C                *************************************************
C                *      UNITS OF PARAMETERS IN THIS PROGRAM       *
C                *                                                *
C                *  1.  DISTANCES = METERS                        *
C                *  2.  WIND SPEEDS = METERS PER SECOND           *
C                *  3.  WIND DIRECTIONS = DIRECTION FROM          *
C                *  4.  TEMPERATURE = DEGREES CELSIUS             *
C                *  5.  RELATIVE HUMIDITY = %, NOT DECIMAL        *
C                *  6.  GRAVITY = METERS/SECOND/SECOND            *
C                *  7.  TIME = HOURS = ZULU TIME                  *
C                *  8.  OMEGA=MILLIBARS/SECOND                    *
C                *                                                *
C                *                                                *
C                *************************************************
C
      DIMENSION Z(24,24,8),SIOLD(24,24),SINEW(24,24),F(24,24),
     1RESID(24,24),SIPROG(24,24),DSI(24,24),DUMMY(24,24),TEVEL(5),
     2TITLE(24),ZBAR(8),VORT(24,24,8),BND(8),BEVEL(8),TEMP(24,24,8),
     3PH(24,24,8),DIR(24,24,8),SPD(24,24,8),USPD(24,24,8),VSPD(24,24,8),
     4OMEGA(24,24,8),D1(22,22),D2(22,22),T5(24)
      DATA BEVEL/'950','850','700','500','300','200','150','100'/
      DATA ZBAR/5.40E2,1.460E3,3.010E3,5.570E3,9.160E3,11.790E3,
     113.620E3,14.0E3/
      DATA BND/5.0E0,5.0E0,6*10.0E0/
      DATA TEVEL/'1700','1830','2000','2130','2300'/
C   BEVEL IS A DUMMY FIELD SO THAT THE PRESSURE LEVEL CAN BE USED IN
C   THE TITLE ON THE OUTPUT
C   ZBAR IS A FIELD THAT IS THE STANDARD ATMOSPHERE VALUE OF HEIGHT
C   FOR EACH PRESSURE LEVEL
C   BND IS A FIELD TO USE AS BANDWIDTH FOR CONTOURING IN METMAP FOR
C   EACH LEVEL
C   TEVEL IS A DUMMY FIELD SO THAT THE TIME CAN BE USED IN THE
C   TITLE ON THE OUTPUT
      ALPHA=1.4
C   ALPHA = OVER-RELAXATION FACTOR
      FBAR=.836E-4
C   FBAR = CORIOLIS PARAMETER FOR LATITUDE = 35.0
```

```
      PI=3.141592
      DIST=1.853E3*5.0
C   DIST=DY=DX=GRID DISTANCE IN METERS
      NK = 7
C   NK = NUMBER OF LEVELS IN THE VERTICAL
      NK1=NK-1
      NK2=NK-2
      NI=24
C   NI = NUMBER OF GRID POINTS ALONG THE I-AXIS
      NI1=NI-1
      NI2=NI-2
      NJ=24
C   NJ = NUMBER OF GRID POINTS ALONG THE J-AXIS
      NJ1=NJ-1
      NJ2=NJ-2
      PHI=35.0*PI/180.
      TWOPHI=2.0*PHI
      FMAP=1.86603/(1.0+SIN(PHI))
      G=9.780356*(1.0+.0052885*(SIN(PHI))**2-.0000059*(SIN(TWOPHI))**2)
C   G=GRAVITY    FMAP=MAP FACTOR FOR 35 DEGREES LATITUDE
      READ(5,6)   (((   Z(I,J,K),I=1,NI),J=1,NJ),K=1,NK2)
      READ(5,5)   (((   Z(I,J,K),I=1,NI),J=1,NJ),K=NK1,NK)
    5 FORMAT(12F6.0)
    6 FORMAT(12F6.1)
      READ(5,6)   (((TEMP(I,J,K),I=1,NI),J=1,NJ),K=1,NK)
      READ(5,6)   (((  RH(I,J,K),I=1,NI),J=1,NJ),K=1,NK)
      READ(5,6)   ((( DIR(I,J,K),I=1,NI),J=1,NJ),K=1,NK)
      READ(5,6)   ((( SPD(I,J,K),I=1,NI),J=1,NJ),K=1,NK)
      DO 8 K=1,NK
      DO 8 I=1,NI
      DO 8 J=1,NJ
      DSI(I,J)=0.0
      USPD(I,J,K) = -SPD(I,J,K)*SIN(DIR(I,J,K)/57.2958)
      VSPD(I,J,K) = -SPD(I,J,K)*COS(DIR(I,J,K)/57.2958)
    8 CONTINUE
C   NOW HAVE   ALL  FIELDS READ IN FOR NK LEVELS IN THE VERTICAL
      DO 30 K=1,NK
      DO 20 I=2,NI1
      DO 20 J=2,NJ1
      DUMMY(I,J)=(1.0/(2.0*DIST))*(USPD(I+1,J,K)-USPD(I-1,J,K)+
     1VSPD(I,J+1,K)-VSPD(I,J-1,K))
C   HERE, DUMMY USED TO REPRESENT DIVERGENCE FIELDS
      DSI(I,J)=DSI(I,J)+DUMMY(I,J)
C   HERE, DSI USED TO REPRESENT THE SUM OF THE DIVERGENCE FIELDS IN A
C   COLUMN FROM 950MB TO 150MB
   20 CONTINUE
   30 CONTINUE
      NUMB=NI2*NJ2
```

```
C   NUMB = NUMBER CF POINTS TO BE RELAXED TO GET PSI FIELD            MN00970
       EPS=58612.                                                     MN00980
C   EPS IS THE CONVERGENCE CRITERIA FOR RELAXATION                    MN00990
       TIME=17.0                                                      MN01000
       ITM=1                                                          MN01010
C   ITM WILL BE USED AS SUBSCRIPT FOR TEVEL                           MN01020
       READ(5,35) TITLE                                               MN01030
C   TITLE IS TITLE INFORMATION FOR METMAP OF HEIGHTS                  MN01035
    35 FORMAT(20A4)                                                   MN01040
       READ(5,35)  T5                                                 MN01050
C   T5 IS TITLE INFORMATION FOR METMAP OF OMEGAS                      MNC1055
       CALL VERTMO(USPD,VSPD,OMEGA,NI,NJ,NK,DIST,DSI)                 MN01060
       CBND=40./3600.0                                                MN01070
C   CBND = BANDWIDTH FOR CONTOURING OF OMEGA FIELD IN METMAP          MN01080
    41 DO 300 K=1,NK                                                  MN01090
C   THIS DO LOOP OBTAINS 1.5 HOUR FORECASTS FOR PSI FOR ALL SEVEN LEVELS MN01100
       TITLE(7) = BEVEL(K)                                            MN01110
       T5(7)=BEVEL(K)                                                 MN01120
       TITLE(23)=TEVEL(ITM)                                           MN01130
C   NEXT, CONVERT OMEGA TO 2-D FIELD FOR METMAP                       MN01135
       DO 42 I=1,22                                                   MN01140
       DO 42 J=1,22                                                   MN01150
       D1(J,I)=OMEGA(I+1,24-J,K)                                      MN01160
    42 CONTINUE                                                       MN01170
       CALL METMAP(D1,22,22,T5,CBND,3.6E0,0.0,0.0,0,1)                MN01180
C   NOW GET PSI FIELD FIRST GUESS FROM THE INITIAL OBSERVED HEIGHTS   MN01190
       DO 55 I=1,NI                                                   MN01200
       DO 55 J=1,NJ                                                   MN01210
       DUMMY(J,I)=Z(I,NJ-J+1,K)                                       MN01220
C   DUMMY IS USED TO HAVE A 2-D FIELD TO TAKE INTO METMAP             MN01230
       SIOLD(I,J)=(G/FBAR)*Z(I,J,K)                                   MN01240
       SINEW(I,J)=SIOLD(I,J)                                          MN01250
       DSI(I,J) = Z(I,J,K)                                            MN01260
C   DSI IS USED TO HAVE A 2-D FIELD TO TAKE INTO RELAXI               MN01270
       SIPROG(I,J)=SIOLD(I,J)                                         MN01280
    55 CONTINUE                                                       MN01290
C   NOW USE METMAP TO CONTOUR THE ORIGINAL HEIGHT FIELD               MN01295
       CALL METMAP(DUMMY,24,24,TITLE,BND(K),.001,0.0,ZBAR(K),0,1)     MN01300
C   NOW RELAX THE FIRST GUESS PSI FIELD                               MN01310
       DO 60 ITER=1,200                                               MN01320
       L = 0                                                          MN01330
       CALL RELAXI(24,24,F,SIOLD,DSI,L,EPS,1.4,FMAP,FBAR,DIST,2,2,ITER, MN01340
      1USPD,VSPD,K,G)                                                 MN01350
       IF(L.GE.NUMB) GO TO 64                                         MN01360
    60 CONTINUE                                                       MN01370
       WRITE(6,62)                                                    MN01380
    62 FORMAT(2X,'200 PASSES AND N O  C O N V E R G E N C E')         MN01390
       GO TO 999                                                      MN01400
```

```
   64 WRITE(6,65)  BEVEL(K),L,ITER,TIME
   65 FORMAT('1',' AT ',A4,' MB    ',I5,'  POINTS WERE WITHIN LIMITS (SHO
     1ULD BE 484) ON PASS NUMBER',I5,'  AT TIME=',F7.2,//)
C
C NOW SET UP THE EQUATIONS WHICH LEAD TO THE PROGGED PSI FIELD
      NUMB=(NI-4)*(NJ-4)
      IZ=1
C  IZ = COUNTER WHICH REPRESENTS NUMBER OF TIME STEPS
      DT=180.
C  DT = FORWARD TIME STEP = .050 HOURS =180 SECONDS
C  NOW MAKE FIRST GUESS AT THE DELTA PSI FIELD
      DO 75 I=1,NI
      DO 75 J=1,NJ
      DSI(I,J)=0.0
   75 CONTINUE
C NOW COMPUTE THE FORCING FUNCTION FOR THE PROG
      CALL FORFUN(VORT,F,FMAP,FBAR,DIST,DT,USPD,VSPD,K,IZ,SIOLD,OMEGA)
C  NOW RELAX THE DELTA PSI FIELD FOR FIRST TIME STEP
      DO 90 ITER=1,100
      L=0
      CALL RELAX(NI1,NJ1,RESID,F,DSI,L,EPS,ALPHA,FMAP,3,3)
      IF(L.GE.NUMB)  GO TO 100
   90 CONTINUE
  100 DO 110 I=3,NI2
      DO 110 J=3,NJ2
      SIPROG(I,J)=SIOLD(I,J)+DSI(I,J)
      SINEW(I,J)=SIPROG(I,J)
  110 CONTINUE
      TIME=TIME+DT/3600.
      DO 210 IZ=2,30
C
C  STARTED AT IZ=2 BECAUSE PROG FOR  TIME=1 ALREADY OBTAINED ABOVE
C  30 TIME STEPS ARE NEEDED FOR A 1.5 HOUR PROG
C
C  UTILIZING THE LATEST PROG,A NEW FORCING FUNCTION FIELD
C  IS NEEDED BEFORE OBTAINING A NEW DELTA PSI FIELD
C
      CALL FORFUN(VORT,F,FMAP,FBAR,DIST,DT,USPD,VSPD,K,IZ,SINEW,OMEGA)
      DO 150 ITER=1,50
      L=0
      CALL RELAX(NI1,NJ1,RESID,F,DSI,L,EPS,ALPHA,FMAP,3,3)
      IF(L.GE.NUMB) GO TO 180
  150 CONTINUE
C  NOW HAVE A NEW DELTA PSI FIELD FOR THE ''IZ'TH'' TIME STEP
  180 TIME=TIME+DT/3600.
      DO 182 I=3,NI2
      DO 182 J=3,NJ2
      SIPROG(I,J)=SIOLD(I,J)+2.0*DSI(I,J)
```

```
  182 CONTINUE                                                          MN01890
      IF(IZ.EQ.10.OR.IZ.EQ.20.OR.IZ.EQ.30)  GO TO 186                   MN01900
C  THIS STEP SMOOTHS EVERY HALF HOUR (10 TIME STEPS) TO KEEP FROM       MN01910
C  HAVING TWO INDEPENDENT SOLUTIONS FROM LEAP-FROG SCHEME               MN01920
      GO TO 195                                                         MN01930
  186 DO 188 I=3,NI2                                                    MN01940
      DO 188 J=3,NJ2                                                    MN01950
      PESID (I,J)=SINEW(I,J)+DSI(I,J)                                   MN01960
C  HERE RESID USED AS DUMMY VARIABLE FOR SIPROG                         MN01970
      SIPROG(I,J)=(SIPROG(I,J)+RESID(I,J))/2.0                          MN01980
  188 CONTINUE                                                          MN01990
  190 CALL SMOOTH(SIPROG,.04,1,3,22,3,22)                               MN02000
  195 DO 200 I=3,NI2                                                    MN02010
      DO 200 J=3,NJ2                                                    MN02020
      SIOLD(I,J)=SINEW(I,J)                                             MN02030
      SINEW(I,J)=SIPROG(I,J)                                            MN02040
  200 CONTINUE                                                          MN02050
  210 CONTINUE                                                          MN02060
      ITM=ITM+1                                                         MN02065
C                                                                       MN02070
C  THIS COMPLETES THE PSI PROG FOR 1.5 HOURS FOR THE K'TH LEVEL         MN02080
C                                                                       MN02090
C  MUST RECOVER Z FIELD  EACH 1.5 HOURS   IN ORDER TO USE CLOUD SUB.    MN02100
C                                                                       MN02110
C  MAKE INITIAL Z-FIELD GUESS                                           MN02120
C                                                                       MN02130
      DO 220 I=3,NI2                                                    MN02140
      DO 220 J=3,NJ2                                                    MN02150
      Z(I,J,K )=(FBAR/G)*SINEW(I,J)                                     MN02160
  220 CONTINUE                                                          MN02170
C  NEXT, SET THE BOUNDARY CONDITIONS                                    MN02180
      DO 225 J=1,NJ                                                     MN02190
      Z(2,J,K)=Z(3,J,K)-(Z(4,J,K)-Z(3,J,K))                             MN02200
      Z    (1,J,K)= Z   (2,J,K)-(Z    (3,J,K)-Z    (2,J,K))             MN02210
      Z(23,J,K)=Z(22,J,K)-(Z(21,J,K)-Z(22,J,K))                         MN02220
      Z    (NI,J,K)= Z   (NI1,J,K)-( Z   (NI1-1,J,K)- Z   (NI1,J,K))    MN02230
  225 CONTINUE                                                          MN02240
      DO 230 I=1,NI                                                     MN02250
      Z(I,2,K)=Z(I,3,K)-(Z(I,4,K)-Z(I,3,K))                             MN02260
       Z   (I,1,K)= Z   (I,2,K)-( Z   (I,3,K)- Z   (I,2,K))             MN02270
      Z(I,23,K)=Z(I,22,K)-(Z(I,21,K)-Z(I,22,K))                         MN02280
      Z    (I,NJ,K)= Z   (I,NJ1,K)-( Z   (I,NJ1-1,K)- Z   (I,NJ1,K))    MN02290
  230 CONTINUE                                                          MN02300
C  NOW RELAX THE PROGGED Z-FIELD                                        MN02310
      NUMB = NI2*NJ2                                                    MN02320
      EPS=0.5                                                           MN02330
C  SET UP THE FORCING FUNCTION FIELD                                    MN02340
      DO 235 I=1,NI                                                     MN02350
```

```
      DO 235 J=1,NJ                                                     MN02360
      DSI(I,J) = Z(I,J,K)                                               MN02370
C  DSI IS BEING USED FOR Z TO HAVE 2-D FIELD FOR RELAX                  MN02380
  235 CONTINUE                                                          MN02390
      DO 240 I=2,NI1                                                    MN02395
      DO 240 J=2,NJ1                                                    MN02400
      T1=SINEW(I+1,J)+SINEW(I-1,J)+SINEW(I,J+1)+SINEW(I,J-1)-4.0*       MN02405
     1SINEW(I,J)                                                        MN02410
      T2= 2.0/(4.0*FBAR)                                                MN02415
      T3=(USPD(I+1,J,K)-USPD(I-1,J,K))*(VSPD(I,J+1,K)-VSPD(I,J-1,K))    MN02420
      T4=(VSPD(I+1,J,K)-VSPD(I-1,J,K))*(USPD(I,J+1,K)-USPD(I,J-1,K))    MN02425
      F(I,J)=(FBAR/G)*(T1+T2*(T3-T4))                                   MN02430
  240 CONTINUE                                                          MN02440
      DO 250 ITER=1,200                                                 MN02450
      L=0                                                               MN02460
      CALL RELAX(NI,NJ,RESID,F,DSI,L,EPS,ALPHA,FMAP,2,2)                MN02470
      IF(L.GE.NUMB) GO TO 251                                           MN02480
  250 CONTINUE                                                          MN02490
  251 DO 252 I=1,NI                                                     MN02500
      DO 252 J=1,NJ                                                     MN02510
      Z(I,J,K) = DSI(I,J)                                               MN02520
  252 CONTINUE                                                          MN02530
C                                                                       MN02540
C    METMAP IS NEXT TO CONTOUR THE Z-FIELD                              MN02550
C                                                                       MN02560
      DO 268 I=1,NI                                                     MN02570
      DO 268 J=1,NJ                                                     MN02580
      DUMMY(J,I)=Z(I,NJ-J+1,K)                                          MN02590
  268 CONTINUE                                                          MN02600
      TITLE(23)=TEVEL(ITM)                                              MN02605
      CALL METMAP(DUMMY,24,24,TITLE,BND(K),.001,0.0,ZBAR(K),0,1)        MN02610
C                                                                       MN02620
C  THIS COMPLETES THE PSI PROG FOR ONE LEVEL(K) AHEAD FOR 1.5 HOURS     MN02630
C                                                                       MN02640
      TIME=TIME-1.5                                                     MN02650
      ITM=ITM-1                                                         MN02660
      EPS=58612.                                                        MN02670
  300 CONTINUE                                                          MN02680
  301 TIME=TIME+1.5                                                     MN02690
      ITM=ITM+1                                                         MN02700
C  THIS COMPLETES ALL THE LEVELS FOR A 1.5 HOUR PROG OF PSI             MN02710
C  NOW PROG AHEAD TEMPERATURE AND RELATIVE HUMIDITY                     MN02720
      CALL PROG(TEMP,RH,DIR,SPD,USPD,VSPD,DIST,NK,NI,NJ,NI1,NJ1,OMEGA)  MN02730
C  NOW SETUP THE PROGGED SOUNDING AND TAKE IT INTO SUBROUTINE CLOUDS    MN02740
      CALL SETUP(TEMP,RH,Z,1,NI,1,NJ)                                   MN02750
      IF(TIME.GT.18.4) GO TO 999                                        MN02760
      GO TO 41                                                          MN02770
  999 STOP                                                              MN02780
```

```
      END                                                               MN02790
      SUBROUTINE VERTMO(U,V,OMEGA,NI,NJ,NK,DIST,DSI)                    V000010
      DIMENSION U(24,24,8),V(24,24,8),OMEGA(24,24,8),DSI(24,24)         V000020
C                                                                       V000030
C   THIS IS A SUBROUTINE TO COMPUTE VERTICAL MOTION ACCORDING TO THE    V000040
C   CONTINUITY EQUATION                                                 V000050
C   BOUNDARY CONDITIONS ARE THAT OMEGA AT 150MB = 0 AND OMEGA AT        V000060
C   1000MB = 0                                                          V000070
C   DSI IS THE NEGATIVE OF THE DIVERGENCE FIELD FOR 1000MB.             V000080
C   THE REASONING FOR THIS IS THAT THE DIVERGENCE AT 1000MB IS SET EQUAL V000090
C   TO THE NEGATIVE OF THE SUM OF THE DIVERGENCES IN THE COLUMN FROM    V000100
C   150MB DOWN THRU 950MB.                                              V000110
C   HERE, OMEGA VALUES ARE COMPUTED FROM 150 MB DOWN THRU 300 MB        V000120
C   AND ARE COMPUTED UPWARDS FROM 1000 MB THRU 700 MB.                  V000130
C   THEN, OMEGA AT 500MB IS COMPUTED AS THE AVERAGE OF THE OMEGA VALUES V000140
C   AT 300MB AND 700MB                                                  V000150
C                                                                       V000160
      NI1 = NI - 1                                                      V000170
      NJ1 = NJ - 1                                                      V000180
      DO 5 I=1,NI                                                       V000190
      DO 5 J=1,NJ                                                       V000200
      OMEGA(I,J,1) = 0.0                                                V000210
      OMEGA(I,J,8) = 0.0                                                V000220
    5 CONTINUE                                                          V000230
      DO 200 K=1,6                                                      V000240
C   FOR THIS DO LOOP, THE K VALUES DO NOT CORRESPOND TO SPECIFIC LEVELS V000250
C   IN THE VERTICAL                                                     V000260
      DO 190 I=2,NI1                                                    V000270
      DO 190 J=2,NJ1                                                    V000280
      IF(K.EQ.1) GO TO 60                                               V000290
      IF(K.EQ.2) GO TO 50                                               V000300
      IF(K.EQ.3) GO TO 40                                               V000310
      IF(K.EQ.4) GO TO 10                                               V000320
      IF(K.EQ.5) GO TO 20                                               V000330
      IF(K.EQ.6) GO TO 30                                               V000340
C                                                                       V000350
C   ******************************************************************* V000360
C                                                                       V000370
C   THE NEXT SET OF STATEMENTS COMPUTES OMEGA AT 200MB AS THE AVERAGE   V000380
C   BETWEEN 175MB AND 225MB                                             V000390
   10 W175=OMEGA(I,J,8)-(25./(4.*DIST))*(U(I+1,J,7)-U(I-1,J,7)+         V000400
     1V(I,J+1,7)-V(I,J-1,7)+U(I+1,J,6)-U(I-1,J,6)+V(I,J+1,6)-V(I,J-1,6)) V000410
C   NOW HAVE OMEGA AT 175MB                                             V000420
C   NEXT GET OMEGA AT 225MB, THEN AVERAGE IT WITH OMEGA AT 175MB        V000430
```

```
C   TO GET OMEGA AT 200MB
       W225=OMEGA(I,J,8)-(75./(4.*DIST))*(U(I+1,J,7)-U(I-1,J,7)+V(I,J+1,
     17)-V(I,J-1,7)+U(I+1,J,5)-U(I-1,J,5)+V(I,J+1,5)-V(I,J-1,5))
       OMEGA(I,J,7) = (W175 + W225)/2.0
C   OMEGA(I,J,7) IS NOW THE OMEGA FOR 200MB
       GO TO 190
C
C   ******************************************************************
C
C   THE NEXT SET CF STATEMENTS COMPUTES OMEGA AT 300MB AS THE AVERAGE
C   BETWEEN 350MB AND 250MB
C   FIRST, CCMPUTE OMEGA AT 350MB
    20 W350=OMEGA(I,J,7)-(150./(4.*DIST))*(U(I+1,J,6)-U(I-1,J,6)+V(I,
     1J+1,6)-V(I, J-1,6)+U(I+1,J,4)-U(I-1,J,4)+V(I,J+1,4)-V(I,J-1,4))
C   NOW COMPUTE OMEGA AT 250MB, THEN AVERAGE IT WITH OMEGA AT
C   350MB TO GET OMEGA AT 300MB
       W250=OMEGA(I,J,7)-(50./(4.*DIST))*(U(I+1,J,6)-U(I-1,J,6)+V(I,J+1,
     16)-V(I,J-1,6)+U(I+1,J,5)-U(I-1,J,5)+V(I,J+1,5)-V(I,J-1,5))
       OMEGA(I,J,6) = (W350 + W250)/2.0
C   OMEGA(I,J,6) IS NOW THE OMEGA FOR 300MB
       GO TO 190
C
C   ******************************************************************
C
C   THE NEXT SET OF STATEMENTS COMPUTES OMEGA AT 500MB AS THE VALUE
C   OBTAINED WHEN THE OMEGA VALUES OF 300MB AND 700MB ARE AVERAGED
    30 OMEGA(I,J,5)=(OMEGA(I,J,6)+OMEGA(I,J,4))/2.0
C   OMEGA(I,J,5) IS NOW THE OMEGA FOR 500MB
       GO TO 190
C
C   ******************************************************************
C
C   THE NEXT SET OF STATEMENTS CCMPUTES OMEGA AT 700MB AS THE
C   AVERAGE BETWEEN OMEGA AT 675MB AND OMEGA AT 725MB
    40 W675=OMEGA(I,J,3)+(175./(4.*DIST))*(U(I+1,J,4)-U(I-1,J,4)+V(I,
     1J+1,4)-V(I,J-1,4)+U(I+1,J,2)-U(I-1,J,2)+V(I,J+1,2)-V(I,J-1,2))
C   NOW COMPUTE OMEGA AT 725MB
       W725=OMEGA(I,J,2)+(225./(4.*DIST))*(U(I+1,J,4)-U(I-1,J,4)+V(I,J+1,
     14)-V(I,J-1,4)+U(I+1,J,1)-U(I-1,J,1)+V(I,J+1,1)-V(I,J-1,1))
       OMEGA(I,J,4) = (W675 + W725)/2.0
C   OMEGA(I,J,4) IS NOW OMEGA AT 700MB
       GO TO 190
C
C   ******************************************************************
C
C   THE NEXT SET OF STATEMENTS CCMPUTES CMEGA AT 850MB AS THE
C   VALUE OBTAINED WHEN VERTICAL MOTIONS AT 825MB AND 900MB ARE AVERAGED
C   USING A WEIGHTED TYPE MEAN
```

```
 50 W825=OMEGA(I,J,2)+(125./(4.*DIST))*(U(I+1,J,3)-U(I-1,J,3)+
   1V(I,J+1,3)-V(I,J-1,3)+U(I+1,J,1)-U(I-1,J,1)+V(I,J+1,1)-V(I,J-1,1))
    W900=OMEGA(I,J,2)+( 50./(4.*DIST))*(U(I+1,J,2)-U(I-1,J,2)+
   1V(I,J+1,2)-V(I,J-1,2)+U(I+1,J,1)-U(I-1,J,1)+V(I,J+1,1)-V(I,J-1,1))
    OMEGA(I,J,3)= (2.0*W825+W900)/3.0
C  OMEGA(I,J,3) IS NOW OMEGA AT 850MB
    GO TO 190
C
C  ********************************************************************
C
C  THE NEXT SET OF STATEMENTS COMPUTES OMEGA AT 950MB AS THE
C  AVERAGE OF THE OMEGA VALUES AT 925MB AND 975MB
C  FIRST COMPUTE OMEGA AT 925MB
 60 W925=OMEGA(I,J,1)+(75.0/2.0)*(((U(I+1,J,2)-U(I-1,J,2)+
   1V(I,J+1,2)-V(I,J-1,2))/(2.0*DIST))-DSI(I,J))
    W975= 0.0+12.5*(((U(I+1,J,1)-U(I-1,J,1)+V(I,J+1,1)-V(I,J-1,1))/
   1(2.0*DIST))-DSI(I,J))
C  NOW AVERAGE W925 AND W975 TO GET OMEGA AT 950MB
    OMEGA(I,J,2) = (W925 + W975)/2.0
C  OMEGA(I,J,2) IS NOW OMEGA AT 950MB
C
C  ********************************************************************
C
190 CONTINUE
200 CONTINUE
C
C  THE THIRD SUBSCRIPT ON OMEGA IS OFF ONE FROM THE THIRD
C  SUBSCRIPT OF THE OTHER PARAMETERS, SO SINCE ALL OMEGAS HAVE NOW
C  BEEN COMPUTED, IT IS NECESSARY TO REVERT THE SUBSCRIPTS BACK SO
C  THEY WILL MATCH
C
    DO 250 K=1,7
    DO 225 I=2,NI1
    DO 225 J=2,NJ1
    OMEGA(I,J,K) = OMEGA(I,J,K+1)*3600.0
225 CONTINUE
C  ALL OMEGAS ARE NOW IN UNITS OF MILLIBARS PER HOUR
    DO 245 I=2,NI1
    DO 245 J=2,NJ1
    OMEGA(I,J,K) = OMEGA(I,J,K)/3600.0
245 CONTINUE
C  OMEGA IS NOW IN MILLIBARS PER SECOND
250 CONTINUE
    RETURN
    END
```

```
      SUBROUTINE FORFUN(VORT,F,FMAP,FBAR,DIST,DT,USPD,VSPD,K,IZ,PSI,     FN00010
     1OMEGA)                                                             FN00020
      DIMENSION VORT(24,24,8),F(24,24),USPD(24,24,8),VSPD(24,24,8),      FN00030
     1OMEGA(24,24,8),PSI(24,24)                                          FN00040
C                                                                        FN00050
C   THIS SUBROUTINE COMPUTES VORTICITY FIELDS THE FIRST TIME IT IS       FN00060
C   CALLED AND COMPUTES THE FORCING FUNCTION EVERY TIME.                 FN00065
C                                                                        FN00070
      IF(K.EQ.1)  DP=100.0                                               FN00080
      IF(K.EQ.2)  DP=250.0                                               FN00090
      IF(K.EQ.3)  DP=350.0                                               FN00100
      IF(K.EQ.4)  DP=400.0                                               FN00110
      IF(K.EQ.5)  DP=300.0                                               FN00120
      IF(K.EQ.6)  DP=150.0                                               FN00130
      IF(K.EQ.7)  DP=50.0                                                FNOC140
      IF(IZ.GE.2) GO TO 215                                              FN00150
      DO 210 KK=1,7                                                      FN00160
      DO 200 I=2,23                                                      FN00170
      DO 200 J=2,23                                                      FN00180
      VORT(I,J,KK)=(FMAP/(2.0*DIST))*(VSPD(I+1,J,KK)-VSPD(I-1,J,KK)-     FN00190
     1USPD(I,J+1,KK)+USPD(I,J-1,KK))                                     FN00200
  200 CONTINUE                                                           FN00210
  210 CONTINUE                                                           FN00220
  215 KK=K                                                               FN00230
C   STATEMENT 215 IS MERELY A DUMMY STATEMENT                            FN00240
      DO 300 I=3,22                                                      FN00250
      DO 300 J=3,22                                                      FN00260
      SIVORT=((FMAP**2)/(DIST**2))*(PSI(I+1,J)+PSI(I-1,J)+PSI(I,J+1)+    FN0027C
     1PSI(I,J-1)-4.0*PSI(I,J))                                           FN00280
C   SIVORT IS THE LAPLACIAN OF THE STREAM FIELD AND IS USED              FNC0290
C   FOR THE VORTICITY FIELD IN THE DIVERGENCE TERM OF THE VORTICITY EQN. FN00300
      IF(K.EQ.1) GO TO 250                                               FN00310
      IF(K.EQ.7) GO TO 275                                               FN00320
      A=OMEGA(I,J,K)*(VORT(I,J,K-1)-VORT(I,J,K+1))/DP                    FN00330
      B=-((FMAP**2.0)/(4.0*DIST**2.0))*((PSI(I,J+1)-PSI(I,J-1))*         FN00340
     1(VORT(I+1,J,K)-VORT(I-1,J,K))-(PSI(I+1,J)-PSI(I-1,J))*             FN00350
     2(VORT(I,J+1,K)-VORT(I,J-1,K)))                                     FN00360
      C=(   SIVORT    +FBAR)*((USPD(I+1,J,K)-USPD(I-1,J,K)+VSPD(I,J+1,K)- FN00370
     1VSPD(I,J-1,K))/(2.0*DIST))                                         FN00380
      D=((OMEGA(I+1,J,K)-OMEGA(I-1,J,K))/(2.0*DIST))*((VSPD(I,J,K-1)-    FN00390
     1VSPD(I,J,K+1))/DP)                                                 FN00400
      E=-((CMEGA(I,J+1,K)-OMEGA(I,J-1,K))/(2.0*DIST))*((USPD(I,J,K-1)-   FN00410
     1USPD(I,J,K+1))/DP)                                                 FN00420
      F(I,J)=((-1.0*DIST**2.0*DT)/(FMAP**2.0))*(A+B+C+D+E)               FN00430
      GO TO 300                                                          FN00440
  250 A=OMEGA(I,J,K)*(VORT(I,J,K  )-VORT(I,J,K+1))/DP                    FN00450
      B=-((FMAP**2.0)/(4.0*DIST**2.0))*((PSI(I,J+1)-PSI(I,J-1))*         FN00460
     1(VORT(I+1,J,K)-VORT(I-1,J,K))-(PSI(I+1,J)-PSI(I-1,J))*             FNC0470
```

```
      2(VORT(I,J+1,K)-VORT(I,J-1,K)))
       C=(   SIVORT    +FBAR)*((USPD(I+1,J,K)-USPD(I-1,J,K)+VSPD(I,J+1,K)-
      1VSPD(I,J-1,K))/(2.0*DIST))
       D=((OMEGA(I+1,J,K)-OMEGA(I-1,J,K))/(2.0*DIST))*((VSPD(I,J,K   )-
      1VSPD(I,J,K+1))/DP)
       E=-((OMEGA(I,J+1,K)-OMEGA(I,J-1,K))/(2.0*DIST))*((USPD(I,J,K   )-
      1USPD(I,J,K+1))/DP)
       F(I,J)=((-1.0*DIST**2.0*DT)/(FMAP**2.0))*(A+B+C+D+E)
       GO TO 300
  275  A=OMEGA(I,J,K)*(VORT(I,J,K-1)-VORT(I,J,K   ))/DP
       B=-((FMAP**2.0)/(4.0*DIST**2.0))*((PSI(I,J+1)-PSI(I,J-1))*
      1(VORT(I+1,J,K)-VORT(I-1,J,K))-(PSI(I+1,J)-PSI(I-1,J))*
      2(VORT(I,J+1,K)-VORT(I,J-1,K)))
       C=(   SIVORT    +FBAR)*((USPD(I+1,J,K)-USPD(I-1,J,K)+VSPD(I,J+1,K)-
      1VSPD(I,J-1,K))/(2.0*DIST))
       D=((OMEGA(I+1,J,K)-OMEGA(I-1,J,K))/(2.0*DIST))*((VSPD(I,J,K-1)-
      1VSPD(I,J,K   ))/DP)
       E=-((OMEGA(I,J+1,K)-OMEGA(I,J-1,K))/(2.0*DIST))*((USPD(I,J,K-1)-
      1USPD(I,J,K   ))/DP)
       F(I,J)=((-1.0*DIST**2.0*DT)/(FMAP**2.0))*(A+B+C+D+E)
  300  CONTINUE
       RETURN
       END


       SUBROUTINE RELAXI(M,N,F,A,B,L,EPS,ALPHA,FMAP,FBAR,DIST,IL,JL,LM,
      1U,V,K,G)
       DIMENSION  A(24,24),B(24,24),F(24,24),R(24,24),U(24,24,8),
      1V(24,24,8)
C   L=NUMBER OF POINTS CONVERGED
C   A=PSI FIELD
C   B=HEIGHT(Z) FIELD
C   R=RESIDUAL
C   F=FORCING FUNCTION
C
       IH=M-1
       JH=N-1
       DO 100 I=IL,IH
       DO 100 J=JL,JH
       W=(U(I+1,J,K)-U(I-1,J,K))*(V(I,J+1,K)-V(I,J-1,K))
       X=(V(I+1,J,K)-V(I-1,J,K))*(U(I,J+1,K)-U(I,J-1,K))
       Z=(G/FBAR)*(B(I+1,J)+B(I-1,J)+B(I,J+1)+B(I,J-1)-4.0*B(I,J))
       Y= A(I+1,J)+A(I-1,J)+A(I,J+1)+A(I,J-1)-4.0*A(I,J)
       R(I,J)=Y+(2.0/(4.0*FBAR))*(W-X)-Z
       IF(ABS(R(I,J)).LT.EPS) L=L+1
       DELTSI=ALPHA*R(I,J)/4.0
```

```
      A(I,J)=A(I,J)+DELTSI                                              RI00220
      F(I,J)=DELTSI                                                     RI00230
  100 CONTINUE                                                          RI00240
      RETURN                                                            RI00250
      END                                                               RI00260


      SUBROUTINE RELAX(M,N,R,F,A,ICOUNT,EPS,ALPHA,FMAP,I1,J1)           RX00010
      DIMENSION R(24,24),F(24,24),A(24,24)                              RX00020
C    ICOUNT = NUMBER OF POINTS CONVERGED                                RX00030
      IHIGH=M-1                                                         RX00040
      JHIGH=N-1                                                         RX00050
      DO 100 I=I1,IHIGH                                                 RX00060
      DO 100 J=J1,JHIGH                                                 RX00070
      R(I,J)=A(I+1,J)+A(I-1,J)+A(I,J+1)+A(I,J-1)-4.0*A(I,J)-F(I,J)      RX00080
      IF(ABS(R(I,J)).LT.EPS) ICOUNT=ICOUNT+1                            RX00090
      A(I,J)=A(I,J)+(ALPHA/4.0)*R(I,J)                                  RX00100
  100 CONTINUE                                                          RX00110
      RETURN                                                            RX00120
      END                                                               RX00130


      SUBROUTINE SMOOTH(D,RF,IPASS,IL,IH,JL,JH)                         SH00010
      DIMENSION D(24,24)                                                SH00020
      DO 450 IK=1,IPASS                                                 SH00030
      DO 400 I=IL,IH                                                    SH00040
      DO 400 J=JL,JH                                                    SH00050
      PL=D(I,J+1)+D(I,J-1)+D(I+1,J)+D(I-1,J)-4.0*D(I,J)                 SH00060
      D(I,J)=D(I,J)+RF*PL                                               SH00070
  400 CONTINUE                                                          SH00080
  450 CONTINUE                                                          SH00090
      RETURN                                                            SH00100
      END                                                               SH00110
```

```
      SUBROUTINE PROG(TEMP,RH,DIR,SPD,USPD,VSPD,DIST,NK,NI,NJ,NI1,NJ1,    PG00010
     1OMEGA)                                                               PG00020
C  THIS SUBROUTINE OBTAINS FORECASTS OF TEMP. AND RELATIVE HUMIDITY       PG00030
      DIMENSION TEMP(24,24,8),RH(24,24,8),DIR(24,24,8),SPD(24,24,8),      PG00040
     1USPD(24,24,8),VSPD(24,24,8),DTEMP(24,24),DRH(24,24),                PG00050
     2OMEGA(24,24,8),DP(7)                                                 PG00060
      DATA DP/-5.0E1,-1.0E2,-1.5E2,-2.0E2,-2.0E2,-1.0E2,-5.0E1/           PG00070
C                                                                          PG00080
C  THIS SUBROUTINE IS DESIGNED TO PROG AHEAD BY TWO METHODS.              PG00090
C  THE FIRST POSSIBILITY IS TO SET THE LOCAL TERM EQUAL TO THE            PG00100
C  NEGATIVE OF THE HORIZONTAL ADVECTION TERM AND THE NEGATIVE OF          PG00110
C  THE VERTICAL ADVECTION TERM.  THE SECOND POSSIBILITY IS TO ONLY        PG00120
C  USE THE NEGATIVE OF THE HORIZONTAL ADVECTION TERM.  IN EITHER          PG00130
C  CASE, ONLY 50% OF THE ADVECTION IS USED.  TO INCLUDE THE VERTICAL      PG00140
C  ADVECTION TERM, TWO CARDS ARE NEEDED BETWEEN THE SECOND DO 50 AND      PG00150
C  THE STATEMENT JUST AFTER IT.                                           PG00160
C                                                                          PG00170
      DELTAT = 90. * 60.                                                   PG00180
C  DELTAT = TIME STEP =1.5 HOURS                                           PG00190
      DO 75 K=1,NK                                                         PG00200
      DO 50 I=2,NI1                                                        PG00210
      DO 50 J=2,NJ1                                                        PG00220
      DTEMP(I,J)=-DELTAT*(USPD(I,J,K)*((TEMP(I+1,J,K)-TEMP(I-1,J,K))/     PG00230
     1(2.*DIST))+VSPD(I,J,K)*((TEMP(I,J+1,K)-TEMP(I,J-1,K))/(2.*DIST)))   PG00240
      DRH  (I,J)=-DELTAT*(USPD(I,J,K)*((  RH(I+1,J,K)-  RH(I-1,J,K))/     PG00250
     1(2.*DIST))+VSPD(I,J,K)*((  RH(I,J+1,K)-  RH(I,J-1,K))/(2.*DIST)))   PG00260
      GO TO 50                                                             PG00270
   20 DTEMP(I,J)=-DELTAT*(USPD(I,J,K)*((TEMP(I+1,J,K)-TEMP(I-1,J,K))/     PG00280
     1(2.*DIST))+VSPD(I,J,K)*((TEMP(I,J+1,K)-TEMP(I,J-1,K))/(2.*DIST))+   PG00290
     2OMEGA(I,J,K)*((TEMP(I,J,K+1)-TEMP(I,J,K))/DP(K+1)))                 PG00300
      DRH  (I,J)=-DELTAT*(USPD(I,J,K)*((  RH(I+1,J,K)-  RH(I-1,J,K))/     PG00310
     1(2.*DIST))+VSPD(I,J,K)*((  RH(I,J+1,K)-  RH(I,J-1,K))/(2.*DIST))+   PG00320
     2OMEGA(I,J,K)*((  RH(I,J,K+1)-  RH(I,J,K))/DP(K+1)))                 PG00330
      GO TO 50                                                             PG00340
   30 DTEMP(I,J)=-DELTAT*(USPD(I,J,K)*((TEMP(I+1,J,K)-TEMP(I-1,J,K))/     PG00350
     1(2.*DIST))+VSPD(I,J,K)*((TEMP(I,J+1,K)-TEMP(I,J-1,K))/(2.*DIST))+   PG00360
     2(OMEGA(I,J,K)/2.0)*(((TEMP(I,J,K+1)-TEMP(I,J,K))/DP(K+1))+          PG00370
     3((TEMP(I,J,K)-TEMP(I,J,K-1))/DP(K))))                               PG00380
      DRH  (I,J)=-DELTAT*(USPD(I,J,K)*((  RH(I+1,J,K)-  RH(I-1,J,K))/     PG00390
     1(2.*DIST))+VSPD(I,J,K)*((  RH(I,J+1,K)-  RH(I,J-1,K))/(2.*DIST))+   PG00400
     2(OMEGA(I,J,K)/2.0)*(((  RH(I,J,K+1)-  RH(I,J,K))/DP(K+1))+          PG00410
     3((  RH(I,J,K)-  RH(I,J,K-1))/DP(K))))                               PG00420
   50 CONTINUE                                                             PG00430
      IF(K.EQ.1)  GO TO 69                                                 PG00440
      DO 55 I=2,NI1                                                        PG00450
      DO 55 J=2,NJ1                                                        PG00460
      TEMP(I,J,K-1)=TEMP(I,J,8)                                            PG00470
      RH(I,J,K-1)=RH(I,J,8)                                                PG00480
```

```
   55 CONTINUE
   69 DO 70 I=2,NI1
      DO 70 J=2,NJ1
      TEMP(I,J,8) = TEMP(I,J,K) +.5* DTEMP(I,J)
      RH(I,J,8) = RH(I,J,K) + .5*DRH(I,J)
      IF(RH(I,J,8).GT.100.) RH(I,J,8) = 100.0
      IF(RH(I,J,8).LT.10.) RH(I,J,8) = 10.0
   70 CONTINUE
   75 CONTINUE
      DO 76 I=2,NI1
      DO 76 J=2,NJ1
      TEMP(I,J,7)=TEMP(I,J,8)
      RH(I,J,7)=RH(I,J,8)
   76 CONTINUE
      DO 100 K=1,NK
C  NEXT, SET THE BOUNDARY CONDITIONS
      DO 80 J=1,NJ
      TEMP(1,J,K)=TEMP(2,J,K)-(TEMP(3,J,K)-TEMP(2,J,K))
      TEMP(NI,J,K)=TEMP(NI1,J,K)-(TEMP(NI1-1,J,K)-TEMP(NI1,J,K))
      RH  (1,J,K)=RH  (2,J,K)-(RH  (3,J,K)-RH  (2,J,K))
      IF(RH( 1,J,K).GT.100.0) RH( 1,J,K)=100.0
      IF(RH( 1,J,K).LT.10.  ) RH( 1,J,K)=10.
      RH  (NI,J,K)=RH  (NI1,J,K)-(RH  (NI1-1,J,K)-RH  (NI1,J,K))
      IF(RH(NI,J,K).GT.100.0) RH(NI,J,K)=100.0
      IF(RH(NI,J,K).LT.10.  ) RH(NI,J,K)=10.
   80 CONTINUE
      DO 85 I=1,NI
      TEMP(I,1,K)=TEMP(I,2,K)-(TEMP(I,3,K)-TEMP(I,2,K))
      TEMP(I,NJ,K)=TEMP(I,NJ1,K)-(TEMP(I,NJ1-1,K)-TEMP(I,NJ1,K))
      RH  (I,1,K)=RH  (I,2,K)-(RH  (I,3,K)-RH  (I,2,K))
      IF(RH(I,1,K).GT.100.0) RH(I,1,K)=100.0
      IF(RH(I, 1,K).LT.10.0) RH(I, 1,K)=10.0
      RH  (I,NJ,K)=RH  (I,NJ1,K)-(RH  (I,NJ1-1,K)-RH  (I,NJ1,K))
      IF(RH(I,NJ,K).GT.100.0) RH(I,NJ,K)=100.0
      IF(RH(I,NJ,K).LT.10.0) RH(I,NJ,K)=10.0
   85 CONTINUE
  100 CONTINUE
C
C  THIS NOW COMPLETES THE PROG FOR   BOTH   PARAMETERS FOR 1.5 HOURS
C
      RETURN
      END
```

```
      SUBROUTINE SETUP(TEMP,RH,Z,IL,IH,JL,JH)
C   THIS SUBROUTINE TAKES THE PROGGED Z-FIELD,TEMP.,AND REL.HUM.,
C   DETERMINES CLOUD BASE AND HAS SUBROUTINE CLOUDS PLOT THE RESULTS OF
C   ANY CONVECTIVE MOTION FOR EACH GRID POINT AT THE SPECIFIED INTERVALS
C   OF TIME AS SPECIFIED IN THE MAIN PROGRAM
      DIMENSION Z(24,24,8),TEMP(24,24,8),RH(24,24,8),P(8),A(8),B(8),
     1C(8)
      NCALL=1
      P(1)=950.
      P(2)=850.
      P(3)=700.
      P(4)=500.
      P(5)=300.
      P(6)=200.
      P(7)=150.
      P(8)=0.0
C   COMPUTE CLOUD BASE,AND CALL CLOUDS FOR EACH GRID POINT
      DO 99 I=IL,IH
      DO 99 J=JL,JH
      IF(I.EQ.10.AND.J.EQ. 1) GO TO 30
      IF(I.EQ.19.AND.J.EQ. 3) GO TO 30
      IF(I.EQ.11.AND.J.EQ. 9) GO TO 30
      IF(I.EQ. 2.AND.J.EQ. 9) GO TO 30
      IF(I.EQ.23.AND.J.EQ. 9) GO TO 30
      IF(I.EQ.15.AND.J.EQ.14) GO TO 30
      IF(I.EQ. 5.AND.J.EQ.17) GO TO 30
      IF(I.EQ.21.AND.J.EQ.18) GO TO 30
      IF(I.EQ.11.AND.J.EQ.23) GO TO 30
      GO TO 99
   30 DZ=200.0
      HEIGHT=Z(I,J,2)
C   HEIGHT = THE HEIGHT OF CLOUD BASE IN METERS
      WRITE(6,35)        TEMP(I,J,2),RH(I,J,2),Z(I,J,2)
   35 FORMAT('1',2X,'BEFORE LCL COMPUTED, TEMP(I,J,2)=',F10.1,3X,
     1'RH(I,J,2)=',F10.1,3X,'AND Z(I,J,2)=',F10.1,/)
C
C   NOW COMPUTE THE CLOUD STRUCTURE IN THE VERTICAL WITH W.D. MODEL
C
      DO 50 KK=1,8
      A(KK)=TEMP(I,J,KK)
      B(KK)=RH(I,J,KK)
      C(KK)=Z(I,J,KK)
   50 CONTINUE
      CALL CLOUDS(P,C,A,B,HEIGHT,DZ,NCALL,I,J)
      NCALL=NCALL+1
   99 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE CLOUDS(PDATA,ZDATA,TDATA,RHDATA,ZZ1 ,DZ,NCALL,IPT,JPT)  CS00010
C                                                                       CS00020
C   SINGLE PRECISICN                                                    CS00030
C                                                                       CS00040
C   W.D. PARAMETERIZED NUMERICAL MODEL CF CUMULUS CONVECTION            CS00050
C   MODIFIED AT NPGS                                                    CS00060
C                                                                       CS00C70
      DIMENSION P(2),T(2),X(2),Z(2),Q(2),QCL(2),QH(2),TVE(2),W(2),U(2), CS00080
     1TH(2),RH(2),DEN(2),PE(200),TE(200),XE(200),ZE(200),UE(200),       CS00090
     2THE(200),RHE(200),AW(200),AQH(200),AQC(200),TC(200),AX(200),      CS0010C
     3UPRAD(200),AO(3),RA(3),DUR(3),ITOP(3),NTF(3),SIZE(3),             CS00110
     4AREA(3),TEMPT(3),TITLE(19),PDATA(8),ZDATA(8),TDATA(8),RHDATA(8)   CS00120
      AP=6958.9262                                                      CS00130
      BP=5.65567                                                        CS00140
      CX=59.01383                                                       CS0015C
C   AP,BP AND CX ARE CONSTANTS USED TO COMPUTE SATURATION VAPOR PRESSURE CS00160
      IF(NCALL.GT.1) GO TO 111                                          CS00170
  100 READ(5,110)   TITLE,JN                                            CS00180
  110 FORMAT(19A4,I4)                                                   CS00190
  111 CONST=273.16                                                      CS00200
C   CONST IS CONVERSION FROM CELSIUS TO KELVIN DEGREES                  CS00210
      RD=287.04                                                         CS00220
      RV=461.5                                                          CS0023C
      RATIO=RD/RV                                                       CS00240
      CP=1004.00                                                        CS00250
C   RD,RV AND CP HAVE UNITS OF JOULES PER KILOGRAM PER DEGREE KELVIN    CS00260
      NUM=1                                                             CSC0270
C   THE R'S ARE GAS CONSTANTS FOR DRY AIR AND WATER VAPOR   CP=SPEC. HEAT CS00280
      PI=3.14159265                                                     CS0029C
      PHI=35.0                                                          CS00300
      PHIRAD= PHI*PI/180.                                               CS00310
      TWOPHI=2.0*PHIRAD                                                 CS00320
      G=9.780356*(1.0+.0052885*( SIN(PHIRAD))**2-.0000059*( SIN(TWOPHI)) CS00330
     1**2)                                                              CS0034C
C   PHI IS MEAN LATITUDE FOR NSSL NETWORK    G=GRAVITY                  CS00350
      CON1=-.62340                                                      CS00360
      CON2=621.7                                                        CS00370
      TNOT=233.16                                                       CS00380
C   CON1=OPTIMUM DIFFERENCE OF SPECIFIC HEATS                           CS00390
C   CON2= LATENT HEAT OF CONDENSATION IN CALORIES/GRAM AT 233 DEGREES   CS00400
  119 WRITE(6,120) TITLE,IPT,JPT                                        CS00410
  120 FORMAT(  /,36X,19A4,//,37X,'GRID POINT IS  I=',I4,4X,'J=',I4)     CS00420
C                                                                       CS00430
C     ****************    PART 1   INTERPOLATION    ******************** CS00440
```

```
C
      WRITE(6,130)
  130 FORMAT(///,60X,16HINITIAL SOUNDING,//)
      N=2
      WRITE(6,140)
  140 FORMAT(20X,'PRESSURE',3X,'HEIGHT',5X,'TEMPERATURE',3X,'RELATIVE HU
     1M',3X,'WIND SPEED',3X,'WIND DIRECTION',/,22X,'(MB)',4X,'(METERS)',
     26X,'(DEG C)',10X,'(%)',8X,'(M/SEC)',6X,'(DIRN. FROM)',//)
      KK=2
      P(1)=PDATA(KK)
      T(1)=TDATA(KK)
      RH(1)=RHDATA(KK)
      U(1)=0.0
      TH(1)=0.0
      Z(1)=ZZ1
  150 FORMAT (2F10.0,10X,4F10.0,F10.0)
      NDZ=DZ
      INDOC=200/NDZ
      TE(1)=T(1)+CONST
      PE(1)=P(1)
      IHT1=Z(1)
      RHE(1)=RH(1)/100.0
      ZE(1)=Z(1)
      UE(1)=U(1)
      THE(1)=TH(1)
      ILEVL=0
      KK = KK + 1
  160 P(2)=PDATA(KK)
      T(2)=TDATA(KK)
      RH(2)=RHDATA(KK)
      U(2)=0.0
      TH(2)=0.0
  170 FORMAT (2F10.0,10X,3F10.0)
      WRITE(6,180) P(1),Z(1),T(1),RH(1),U(1),TH(1)
  180 FORMAT( 19X,F8.2,F11.2,F14.2,F15.1,F13.2,F17.0)
      IF(P(2)) 260,260,190
  190 Z(2)=Z(1)+ALOG(P(1)/P(2))*RD*(T(1)+T(2)+(2.0*CONST))/(2.0*G)
      XLNP=ALOG(P(2)/P(1))
      DTDP=(T(2)-T(1))/XLNP
      DRHDP=(RH(2)-RH(1))/(XLNP*100.0)
      DUDP=(U(2)-U(1))/XLNP
      DTHDP=(TH(2)-TH(1))/XLNP
      DO 240 J=N,200
      IF(ILEVL) 200,200,210
  200 PE(J)=PE(J-1)* EXP(-G*DZ/(RD*TE(J-1)))
      DP=ALOG(PE(J)/PE(J-1))
      TE(J)=TE(J-1)+(DTDP*DP)
      RHE(J)=RHE(J-1)+(DRHDP*DP)
```

```
      UE(J)=UE(J-1)+(DUDP*DP)                                           CS00930
      THE(J)=THE(J-1)+(DTHDP*DP)                                        CS00940
      GO TO 220                                                         CS00950
  210 PE(J)=P(1)* EXP(-G*(ZE(2)-Z(1))/(RD*(T(1)+CONST)))                CS00960
      DP=ALOG(PE(J)/P(1))                                               CS00970
      TE(J)=T(1)+(DTDP*DP)+CONST                                        CS00980
      RHE(J)=RH(1)/100.0+(DRHDP*DP)                                     CS00990
      UE(J)=U(1)+(DUDP*DP)                                              CS01000
      THE(J)=TH(1)+(DTHDP*DP)                                           CS01010
      ILEVL=0                                                           CS01020
  220 ZE(2)=ZE(1)+DZ                                                    CS01030
      IF(ZE(2)-Z(2)) 230,230,250                                        CS01040
  230 ZE(1)=ZE(2)                                                       CS01050
  240 CONTINUE                                                          CS01060
  250 Z(1)=Z(2)                                                         CS01070
      P(1)=P(2)                                                         CS01080
      T(1)=T(2)                                                         CS01090
      U(1)=U(2)                                                         CS01100
      TH(1)=TH(2)                                                       CS01110
      RH(1)=RH(2)                                                       CS01120
      N=J                                                               CS01130
      ILEVL=1                                                           CS01140
      KK=KK+1                                                           CS01150
      GO TO 160                                                         CS01160
  260 IF(JN) 310,310,270                                                CS01170
  270 WRITE(6,280)                                                      CS01180
  280 FORMAT('1',60X,'INTERPOLATED SOUNDING',//)                        CS01190
      WRITE(6,140)                                                      CS01200
  285 FORMAT(20X,'PRESSURE',3X,'HEIGHT',5X,'TEMPERATURE',3X,'RELATIVE HU CS01210
     1M',3X,'WIND SPEED',3X,'WIND DIRECTION',/,22X,'(MB)',4X,'(METERS)', CS01220
     26X,'(DEG K)',10X,'(%)',8X,'(M/SEC)',6X,'(DIRN. FROM)',//)         CS01230
      ZE(1)=IHT1-NDZ                                                    CS01240
      DO 290 I=1,N                                                      CS01250
      ZE(1)=ZE(1)+DZ                                                    CS01260
      TE(I) = TE(I) - CONST                                             CS01270
      WRITE(6,300) PE(I),ZE(1),TE(I),RHE(I),UE(I),THE(I)                CS01280
      TE(I) = TE(I) + CONST                                             CS01290
  290 CONTINUE                                                          CS01300
  300 FORMAT( 19X,F8.2,F11.2,F14.2,2PF15.3,0PF13.2,F17.0)               CS01310
  310 DO 320 I=1,N                                                      CS01320
      ES= EXP(CX-AP/TE(I))/(TE(I)**BP)                                  CS01330
      XS=(RATIO*ES)/(PE(I)-ES)                                          CS01340
      XE(I)=XS*RHE(I)                                                   CS01350
  320 THE(I)=THE(I)*(PI/180.)                                           CS01360
C                                                                       CS01370
C    ************   PART 2   MODEL COMPUTATIONS    ************          CS01380
C                                                                       CS01390
  330 NCR=1                                                             CS01400
```

```
      NBS=1                                                             CS01410
      LWC=0                                                             CS01420
      A1=0.2                                                            CS01430
      AK1=.001                                                          CS01440
      AKF1=.001                                                         CS01450
      AK2=.0052                                                         CS01460
      AKF2=.0052                                                        CS01470
      CRAD=1.0                                                          CS01480
      TF=248.0                                                          CS01490
  340 FORMAT(3I1,F4.0,6F10.0,F2.0)                                      CS01500
  350 SIZE(NUM)=CRAD                                                    CS01510
      AO(NUM)=A1                                                        CS01520
      NT=273.0-TF                                                       CS01530
      ZE(1)=IHT1                                                        CS01540
      WRITE(6,360) AO(NUM),CRAD,AK1,AKF1,AK2,AKF2,TF                    CS01550
  360 FORMAT('1MU=',F3.1,'/RUP',2X,'RUP=',F6.3,' KM',4X,'K1=',F5.3,2X,  CS01560
     1'K1F=',F5.3,2X,'K2=',F6.4,2X,'K2F=',F6.4,4X,'TF=',F6.1)           CS01570
      RAD1=1000.0*CRAD                                                  CS01580
      XMU1=AO(NUM)/RAD1                                                 CS01590
      WRITE(6,370)                                                      CS01600
  370 FORMAT('0',2X,'HEIGHT',4X,'HEIGHT',3X,'PRESSURE',2X,'VERTICAL',   CS01610
     13X,'CLOUD',6X,'TEMP.',4X,'MIXING',4X,'CLOUD',5X,'HYDRO',7X,'Z',   CS01620
     28X,'UPDRAFT')                                                     CS01630
      WRITE(6,380)                                                      CS01640
  380 FORMAT(' ',31X,'VELOCITY',3X,'TEMP.',5X,'EXCESS',4X,'RATIO',5X,   CS01650
     1'WATER',5X,'WATER',5X,'FACTOR',5X,'RADIUS')                       CS01660
      WRITE(6,390)                                                      CS01670
  390 FORMAT(' ',1X,'(METERS)',3X,'(FEET)',5X,'(MB)',5X,'(MPS)',4X,     CS01680
     1'(DEG A)',4X,'(DEG C)',2X,'(GM/KG)',3X,'(GM/KG)',3X,'(GM/KG)',3X, CS01690
     2'(MM6/M3)',4X,'(METERS)',/)                                       CS01700
C                                                                       CS01710
C    *******       INITIALIZATION     *********                         CS01720
C                                                                       CS01730
      IF(NBS) 420,420,410                                               CS01740
  410 J=1                                                               CS01750
      NFRZ=0                                                            CS01760
C   NFRZ=KEY TO INDICATE WHEN FREEZING LEVEL IS REACHED                 CS01770
      TH(1)=THE(1)                                                      CS01780
      U(1)=UE(1)                                                        CS01790
      T(1)=TE(1)                                                        CS01800
      TC(1)=TE(1)                                                       CS01810
C   TC= TEMPERATURE OF CLOUD                                            CS01820
      X(1)=XE(1)/RHE(1)                                                 CS01830
      QH(1)=0.0                                                         CS01840
      AQH(1)=0.0                                                        CS01850
      QCL(1)=0.0                                                        CS01860
      W(1)=0.5                                                          CS01870
C  W=0.5 IS ASSUMPTION OF .5 METERS PER SECOND VERTICAL VELOCITY        CS01880
```

```
      AW(1)=W(1)
      IHT=IHT1
      RO=0.0
      RAD=RAD1
      UPRAD(1)=CRAD
      DTMAX=0.0
      SDTMX=0.0
      JDTMX=0
      JSDTM=0
      GO TO 440
C
C     **************  INITIALIZATION AT PREVIOUS FREEZING LEVEL ********
C
  420 J=ITOB
      DTMAX=SDTMX
      JDTMX=JSDTM
      AREA(NUM)=AREA(NUM-1)
      T(1)=TC(ITOB)
      X(1)=AX(ITOB)
      QCL(1)=AQC(ITOB)
      QH(1)=AQH(ITOB)/1000.
      W(1)=AW(ITOB)
      RAD=UPRAD(ITOB)*RAD1
      RO=ROD
      TH(1)=THTH
      U(1)=UU
      IHT=IHT1
      IST=INDOC+1
      DO 430 I=IST,ITOB,INDOC
      IHT=IHT+NDZ*INDOC
      IHTT=(IHT*328)/100
      A=TC(I)-TE(I)
      PRAD=UPRAD(I)*RAD1
      BQH=AQH(I)/1000.
      C=(14000.*BQH)**1.136
      ZFZC=200.*(C**1.6)
  430 WRITE(6,790) IHT,IHTT,PE(I),AW(I),TC(I),A,AX(I),AQC(I),BQH,ZFZC,
     1PRAD
  440 TVE(1)=T(1)*(1.0+.61*X(1))
      Q(1)=QCL(1)+QH(1)
      DEN(1)=PE(J)/(RD*TE(J))*.1
      INDIC=0
      DA=0.0
C     DA= HEAT ADDED DUE TO FREEZING OF LIQUID WATER
      AP=6958.9262
      BP=5.65567
      CX=59.01383
      XK4=15.39
```

```
      AKA=AK2
      AK=AK1
      UBASE=UE(1)
      THBASE=THE(1)
      IT=0
C   IT IS A KEY TO INDICATE WHEN ICE NUCLEATION LEVEL REACHED
  450 J=J+1
      IF(IT) 451,451,459
  451 XL=CON1*(T(1)-TNOT)+CON2
      XL=XL*4.186E3
C  THIS CONVERTS XL FROM CALORIES/GRAM TO JOULES/KILOGRAM
  459 IF(N-J) 810,460,460
  460 XMU=AC(NUM)/RAD
      IHT=IHT+NDZ
      INDIC=INDIC+1
C
C    **********   MOIST OR ICE ADIABATIC ASCENT   **********
C
      A=-G*DZ/CP
      B=1.0+((X(1)*XL)/(RD*T(1)))
      C=1.0+XL*XL*X(1)/(CP*RV*T(1)**2)
C  SEE EQN. 190 PAGE 377 OF HANDBOOK OF METEOROLOGY
      T(2)=T(1)+(A*B+DA)/C
      IF(IT) 464,464,465
  464 XL=CON1*(T(2)-TNOT)+CON2
      XL=XL*4.186E3
C
C    **********   MIXING AT CONSTANT PRESSURE   **********
C
  465 ES= EXP(CX-AP/ T(2))/( T(2)**BP)
      X(2)=(RATIO*ES)/(PE(J)-ES)
      A=(XMU1*DZ*XL/CP)*(X(2)-XE(J))
      B=(XMU*DZ)*(T(2)-TE(J))
      C=1.0+XL*XL*X(2)/(CP*RV*T(2)**2)
C  A,B, AND C ARE NOW BEING REDEFINED (EQN.I P.19 OF W.D. PAPER)
      T(2)=T(2)-(A+B)/C
      ES= EXP(CX-AP/ T(2))/( T(2)**BP)
      X(2)=(RATIO*ES)/(PE(J)-ES)
C
C    **********   CLOUD PHYSICS TERMS   **********
C
      DENS=PE(J)/(RD*TE(J))*1.0E5
      AA=0.5000
      ADEN=AA/DENS
      IF(QCL(1)-ADEN)  470,470,480
  470 A=0.0
      GO TO 490
C
```

```
C     **********   CONVERSION    **********                        CS02850
C                                                                  CS02860
  480 A=(QCL(1)-ADEN)*AK                                           CS02870
C                                                                  CS02880
C     **********   COLLECTION    **********                        CS02890
C                                                                  CS02900
  490 B=423.0*AKA*QCL(1)*QH(1)**.875                               CS02910
      A=(A/W(1)+B/(W(1)-XK4*QH(1)**.125))*DZ                       CS02920
      QCL(2)=QCL(1)-(X(2)-X(1))-A                                  CS02930
      IF(QCL(2)) 500,510,510                                       CS02940
  500 QCL(2)=0.0                                                   CS02950
      QH(2)=QH(1)                                                  CS02960
      GO TO 550                                                    CS02970
  510 QCL(2)=QCL(2)-XMU1*DZ*(X(2)-XE(J)+QCL(1))                    CS02980
      IF(QCL(2)) 520,530,530                                       CS02990
  520 QCL(2)=0.0                                                   CS03000
  530 QH(2)=QH(1)+A                                                CS03010
      IF(QH(2)) 540,550,550                                        CS03020
  540 QH(2)=0.0                                                    CS03030
  550 IF(LWC) 560,560,570                                          CS03040
C  LWC IS A KEY TO INDICATE THE MANNER BY WHICH BUOYANCY SHOULD BE DECR. CS03050
  560 Q(2)=QCL(2)+QH(2)                                            CS03060
      WATER=Q(2)                                                   CS03070
      GO TO 590                                                    CS03080
  570 IF(W(1)-XK4*QH(1)**.125) 580,580,560                         CS03090
  580 Q(2)=QCL(2)+QH(2)                                            CS03100
      WATER=QCL(2)                                                 CS03110
  590 TVC=T(2)*(1.0+.61*X(2))                                      CS03120
      TVC1=T(1)*(1.0+.61*X(1))                                     CS03130
      TVBAR=(TVC1+TVC )/2.0                                        CS03140
      TVE(2)=TE(J)*(1.0+.61*XE(J))                                 CS03150
C                                                                  CS03160
C     **********   VERTICAL VELOCITY COMPUTATION   **********      CS03170
C                                                                  CS03180
      D=(TVE(2)+TVE(1))*.5                                         CS03190
      A=TVBAR-D                                                    CS03200
      B=G*DZ*((A/D)-WATER)                                         CS03210
      A=(1.0-2.0*XMU*DZ)*W(1)**2+2.0*B                             CS03220
  600 IF(A) 610,670,670                                            CS03230
  610 IF(T(2)-TF) 630,630,620                                      CS03240
  620 NFRZ=1                                                       CS03250
      GO TO 810                                                    CS03260
  630 IF(IT) 640,640,810                                           CS03270
  640 DEN(2)=PE(J)/(RD*TE(J))*.1                                   CS03280
      ROD=RO+(QH(2)+QH(1))*(DEN(2)+DEN(1))*.25*DZ                  CS03290
      ITOB=J-1                                                     CS03300
      IF(JDTMX-ITOB) 650,650,660                                   CS03310
  650 SDTMX=DTMAX                                                  CS03320
```

```
      JSDTM=JDTMX                                                       CS03330
  660 UU=U(1)                                                           CS03340
      THTH=TH(1)                                                        CS03350
      GO TO 810                                                         CS03360
  670 W(2)=A**.5                                                        CS03370
C                                                                       CS03380
C     ***************   FREEZING   ***************                      CS03390
C                                                                       CS03400
      IF(IT) 680,680,720                                                CS03410
  680 IF(T(2)-TF) 690,690,720                                           CS03420
  690 IT=IT+1                                                           CS03430
      AP=6151.0205                                                      CS03440
      PP=0.0                                                            CS03450
      CX=24.3277                                                        CS03460
      XK4=11.58                                                         CS03470
      AK=AKF1                                                           CS03480
      AKA=AKF2                                                          CS03490
      DES=1.747E-1                                                      CS03495
      DA=(2.834E6-XL)*Q(2)/CP+(DES*RATIO*2.834E6/(PE(J)*CP))            CS03500
      XL=2.834E6                                                        CS03510
      ITOB=J-1                                                          CS03520
      IF(JDTMX-ITOB) 700,700,710                                        CS03530
  700 SDTMX=DTMAX                                                       CS03540
      JSDTM=JDTMX                                                       CS03550
  710 UU=U(1)                                                           CS03560
      THTH=TH(1)                                                        CS03570
      ROD=RO                                                            CS03580
      GO TO 460                                                         CS03590
  720 DA=0.0                                                            CS03600
C                                                                       CS03610
C     ************   UPDRAFT RADIUS   ************                      CS03620
C                                                                       CS03630
      DEN(2)=PE(J)/(RD*TE(J))*.1                                        CS03640
      IF(NCR) 730,730,740                                               CS03650
  730 RAD=RAD*((DEN(1)/DEN(2))*(W(1)/W(2)))**.5                         CS03660
C                                                                       CS03670
C     ************   TOTAL PRECIPITATION   ********                     CS03680
C                                                                       CS03690
  740 RO=RO+(QH(2)+QH(1))*(DEN(2)+DEN(1))*.25*DZ                        CS03700
C  RO=TOTAL RAINFALL.   RO=0 AT CARD CS01910                            CS03710
C                                                                       CS03720
C     ************   RADAR REFLECTIVITY FACTOR   *************          CS03730
C                                                                       CS03740
      C=(14000.0*QH(2))**(1.136)                                        CS03750
      ZFZC=200.0*(C**1.6)                                               CS03760
C                                                                       CS03770
C     *************   UPDRAFT AREA   **************                     CS03780
C                                                                       CS03790
```

```
      IF(T(2)-TE(J)-T(1)+TE(J-1)) 750,770,770                           CS03800
  750 IF(DTMAX-T(1)+TE(J-1)) 760,770,770                                CS03810
  760 HT=IHT-IHT1-NDZ                                                   CS03820
      DTMAX=T(1)-TE(J-1)                                                CS03830
      AREA(NUM)=DTMAX/(T(1)-TE(1)+G/CP*HT)                              CS03840
      DENOM=T(1)-TE(1)+G/CP*HT                                          CS03850
      WRITE(6,765)  AREA(NUM),DTMAX,DENOM                               CS03860
  765 FORMAT(2X,3E15.5,3X,'CARD CS03870',50('*'))                       CS03870
      JDTMX=J                                                           CS03880
  770 IF(INDIC-INDOC) 800,780,780                                       CS03890
  780 INDIC=0                                                           CS03900
C                                                                       CS03910
C    **********  TABULAR OUTPUT OF PROFILES  ************                CS03920
C                                                                       CS03930
      D=T(2)-TE(J)                                                      CS03940
      IHTFT=(IHT*328)/100                                               CS03950
      WRITE(6,790) IHT,IHTFT,PE(J),W(2),T(2),D,X(2),QCL(2),QH(2),ZFZC,  CS03960
     1RAD                                                               CS03970
  790 FORMAT(I8,I10,  F11.4,  F10.5,2F10.4,3P3F10.6,2X,0PE10.2,F10.2)   CS03980
C                                                                       CS03990
C    ************  STORE PROFILES FOR GRAPHICAL OUTPUT  **********       CS04000
C                                                                       CS04010
  800 ITAB=J                                                            CS04020
      AW(ITAB)=W(2)                                                     CS04030
      TC(ITAB)=T(2)                                                     CS04040
      AQH(ITAB)=QH(2)*1000.0                                            CS04050
      AX(ITAB)=X(2)                                                     CS04060
      AQC(ITAB)=QCL(2)                                                  CS04070
      UPRAD(ITAB)=RAD/RAD1                                              CS04080
C                                                                       CS04090
C    *******  PREPARE FOR NEXT GRID STEP  *********                     CS04100
C                                                                       CS04110
      Q(1)=Q(2)                                                         CS04120
      TVE(1)=TVE(2)                                                     CS04130
      T(1)=T(2)                                                         CS04140
      X(1)=X(2)                                                         CS04150
      QCL(1)=QCL(2)                                                     CS04160
      QH(1)=QH(2)                                                       CS04170
      DEN(1)=DEN(2)                                                     CS04180
      W(1)=W(2)                                                         CS04190
      GO TO 450                                                         CS04200
C                                                                       CS04210
C    **********   TOTAL PRECIPITATION   ***************                 CS04220
C                                                                       CS04230
  810 PA(NUM)=RO*39.37                                                  CS04240
C                                                                       CS04250
C    **********   DURATION OF PRECIPITATION  **********                 CS04260
C                                                                       CS04270
```

```
      A=IHT-IHT1                                                        CS04280
      IF(QH(2).GE..1E-5) GO TO 815                                      CS04290
      QH(2)=.1E-5                                                       CS04300
      WRITE(6,814)                                                      CS04310
  814 FORMAT(2X,'CARD CS04320 ',50('*'))                                CS04320
  815 DUR(NUM)=(A/(XK4*QH(2)**.125))/60.0                               CS04330
C                                                                       CS04340
C    **********   CLOUD TOP HEIGHT      ***************                 CS04350
C                                                                       CS04360
      ITOP(NUM)=IHT                                                     CS04370
C                                                                       CS04380
C    **********   FREEZING TEMPERATURE (DEG. C) *******                 CS04390
C                                                                       CS04400
      NTF(NUM)=-NT                                                      CS04410
C                                                                       CS04420
C    **********   CLOUD TOP TEMPERATURE (DEG. C) ******                 CS04430
C                                                                       CS04440
      TEMPT(NUM)=T(2)-CONST                                             CS04450
      WRITE(6,820) RA(NUM),DUR(NUM),ITOP(NUM),AREA(NUM)                 CS04460
  820 FORMAT(1X,'TOTAL RAIN=',F10.4,' INCHES PER CLOUD',5X,'RAIN LASTS', CS04470
     1F10.2,' MINUTES',5X,'CLOUD TOP=',I5,'METERS',3X,'UPDRAFT AREA=',  CS04480
     22PF6.1,' %',//)                                                   CS04490
C                                                                       CS04500
C    **** IF TOP DOES NOT REACH WARMEST FREEZING LEVEL, PROCEED TO  **** CS04510
C    **** NEXT BOUNDARY CONDITION CARD                              **** CS04520
C                                                                       CS04530
  830 NCR=1                                                             CS04540
      NBS=0                                                             CS04550
      LWC=0                                                             CS04560
      A1=0.2                                                            CS04570
      AK1=.001                                                          CS04580
      AKF1=.001                                                         CS04590
      AK2=.0052                                                         CS04600
      CRAD=1.0                                                          CS04610
      TF=248.0                                                          CS04620
C                                                                       CS04630
      IF(NBS) 840,840,870                                               CS04640
  840 IF(NFRZ) 870,870,850                                              CS04650
  850 WRITE(6,860) AO(NUM),CRAD,AK1,AKF1,AK2,AKF2,TF                    CS04660
  860 FORMAT(1X,'MU=',F3.1,'/RUP',2X,'RUP=',F5.3,' KM',4X,'K1=',F5.3,2X, CS04670
     1'K1F=',F5.3,2X,'K2=',F6.4,2X,'K2F=',F6.4,4X,'TF=',F6.1,2X,        CS04680
     2'TOP TEMP. ABOVE TF')                                             CS04690
      GO TO 900                                                         CS04700
C                                                                       CS04710
C    *****  GRAPHICAL OUTPUT OF SELECTED PROFILES  **********            CS04720
C                                                                       CS04730
  870 CALL GRPHCL(ITOP(NUM),NUM,NDZ,INDOC,ITAB,TC,TE,AW,AQH,UPRAD,AO(NUM CS04740
     1))                                                                CS04750
```

```
      WRITE(6,820) RA(NUM),DUR(NUM),ITOP(NUM),AREA(NUM)
  900 RETURN
      END


      SUBROUTINE GRPHCL(ITOP,NUM,NDZ,INDOC,ITAB,TC,TE,AW,AQH,UPRAD,A1)
C   SUBROUTINE TO GRAPH SELECTED PROFILES. IN THIS CASE PROFILES OF
C   VERTICAL VELOCITY,TEMPERATURE EXCESS,HYDROMETEOR WATER, AND
C   UPDRAFT RADIUS ARE GRAPHED.
      INTEGER*2 CH,CHAR
      DIMENSION TC(200),TE(200),AW(200),AQH(200),UPRAD(200),VAR(10),
     1STRT(10),GI(10),CH(10),CHAR(100)
      DATA CH/'W','T','Q','$','R',5*' '/
      DATA STRT/-5.0E-1,-1.02E1,-1.0E-1,-1.02E1,-3.3333E-2,5*0.0E0/
      DATA GI/5.0E0,2.0E0,1.0E0,2.0E0,3.3333E-1,5*0.0E0/
      WRITE(6,5)  ITAB,NUM,NDZ,A1
    5 FORMAT('1',10X,'G R A P H I C A L   D I S P L A Y   NUMBER OF LEVEL
     1S =',I5,//,11X,'BOUNDARY CONDITICN NUMBER',I4,5X,'VERTICAL GRID DI
     2STANCE=',I4,1X,'METERS', 5X,'MU=',F5.2,'/RADIUS',///)
      IHT=ITOP+NDZ*INDOC-NDZ
      CALL SCALE(5,STRT,GI)
      DO 10 I=1,ITAB,INDOC
      IHT=IHT-NDZ*INDOC
      J=ITAB-I+1
      VAR(1)=AW(J)
      VAR(2)=TC(J)-TE(J)
      VAR(3)=AQH(J)
      VAR(4)=0.0
      VAR(5)=UPRAD(J)
      CALL GRAPH(5,CHAR,VAR,CH)
   10 WRITE(6,20) IHT,CHAR
   20 FORMAT(1HS,I6,100A1)
      WRITE(6,30)
   30 FORMAT(1H ,7X,11('*',9X))
      WRITE(6,40)
   40 FORMAT(1H ,6X,102('*'))
      WRITE(6,50) (I,I=5,50,5)
   50 FORMAT(1H ,7X,'0',10(I10),' W(M/SEC)')
      WRITE(6,60) (I,I=2,10,2)
   60 FORMAT(1H ,5X,'-10',8X,'-8',8X,'-6',8X,'-4',8X,'-2',9X,'0',5(I10),
     1' T EXCESS(C)')
      WRITE(6,70) (I,I=1,10)
   70 FORMAT(1H ,7X,'0',10(I10),' Q HYDRO(G/KG)')
      WRITE(6,80) (I,I=1,3)
   80 FORMAT(  8X,'0',3I30,10X,' UPDRAFT RADIUS(KM)')
      RETURN
```

```
      END                                                               GL00420

      SUBROUTINE GRAPH(N,CHAR,YC,CH)                                    GH00010
      INTEGER*2 CH(10),CHAR(100),Z/' '/                                 GH00020
      DIMENSION YC(10),GI(10),STRT(10),GIMOD(10)                        GH00030
    9 DO 10 I=1,100                                                     GH00040
   10 CHAR(I)=Z                                                         GH00050
      GO TO 30                                                          GH00060
      ENTRY SCALE(N,STRT,GI)                                            GH00070
      DO 20 I=1,N                                                       GH00080
   20 GIMOD(I)=10.0/GI(I)                                               GH00090
      RETURN                                                            GH00100
   30 DO 50 I=1,N                                                       GH00110
      DI=(YC(I)-STRT(I))*GIMOD(I)+1.0                                   GH00120
      IS=DI                                                             GH00130
      IF((DI-IS).GE.0.5)IS=IS+1                                         GH00140
      IF(IS.LE.0) GO TO 45                                              GH00150
   40 IF(IS.LE.100) GO TO 42                                            GH00160
      IS=IS-100                                                         GH00170
      GO TO 40                                                          GH00180
   42 CHAR(IS) = CH(I)                                                  GH00190
      GO TO 50                                                          GH00200
   45 WRITE(6,46)  IS,DI,N,I,YC(I),STRT(I),GIMOD(I)                     GH00210
   46 FORMAT(2X,'IS=',I8,2X,'DI=',E9.1,2X,'N=',I3,2X,'I=',I3,2X,'YC=',  GH00220
     1E9.1,2X,'STRT=',E9.1,2X,'GIMOD=',E9.1)                            GH00230
   50 CONTINUE                                                          GH00240
      RETURN                                                            GH00250
      END                                                               GH00260


      SUBROUTINE METMAP(Y,N,M,T,BND,AZ,BZ,AMIN,IJT,ICON)                MP00010
C   THIS SUBROUTINE OBJECTIVELY CONTOURS THE FIELD BROUGHT IN AS Y      MP00020
      REAL*4 IH,KG,ITJZ                                                 MP00030
      DIMENSION A(140),B(140),C(140),D(140),IH(20),Y(N,M),TP(5),XMT(5), MP00040
     1RT(5),KG(10),T(24)                                                MP00050
C                                                                       MP00060
C                                                                       MP00070
      DATA DUE/4H    /,EPL/4H+   /,EMI/4H-   /,IH/1H0,1H ,1H1,1H ,1H2,  MP00080
     11H ,1H3,1H ,1H4,1H ,1H5,1H ,1H6,1H ,1H7,1H ,1H8,1H ,1H9,1H /,KG/  MP00090
     21H0,1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8,1H9/,BLK/4H    /              MP00100
C                                                                       MP00110
      YMIN=Y(1,1)                                                       MP00120
      YMAX=Y(1,1)                                                       MP00130
```

```
      DO 20 I=1,M                                                       MP00140
      DO 10 J=1,N                                                       MP00150
      YMIN=AMIN1(YMIN,Y(J,I))                                           MP00160
      YMAX=AMAX1(YMAX,Y(J,I))                                           MP00170
   10 CONTINUE                                                          MP00180
   20 CONTINUE                                                          MP00190
      DELY=YMAX-YMIN                                                    MP00200
      IF(BND) 25,25,30                                                  MP00210
   25 BND=DELY/15.0                                                     MP00220
   30 IF (AMIN-YMIN) 31,31,32                                           MP00230
   31 IF (IJT) 33,32,33                                                 MP00240
   32 PD=YMIN/BND                                                       MP00250
      PF=ABS(PD-INT(PD))                                                MP00260
      IF (YMIN) 2,1,1                                                   MP00270
    1 AMIN=YMIN-PF*BND                                                  MP00280
      GO TO 33                                                          MP00290
    2 AMIN=YMIN-(1.0-PF)*BND                                            MP00300
   33 AHLD=AZ                                                           MP00310
      IF(AZ) 55,35,55                                                   MP00320
   35 SM=AMAX1(ABS(YMIN),ABS(YMAX))                                     MP00330
      NS=0                                                              MP00340
   40 NS=NS+1                                                           MP00350
      SM=10.0*SM                                                        MP00360
      IF(SM-1.0)40,50,45                                                MP00370
   45 NS=NS-1                                                           MP00380
      SM=SM/10.0                                                        MP00390
      IF(SM-1.0)50,50,45                                                MP00400
   50 AHLD=10.0**NS                                                     MP00410
   55 HBND=BND/2.0                                                      MP00420
      PRINT 70                                                          MP00430
      PRINT 6,T                                                         MP00440
    6 FORMAT(5X,24A4,//)                                                MP00450
      PRINT 57,AHLD,BZ                                                  MP00460
   57 FORMAT(1H0,65HTHE FOLLOWING TRANSFORMATION WAS PERFORMED ON THE IN MP00470
     1PUT MATRIX      /5X,1H(,E12.5,8H*Y(I,J)+,E12.5,1H) //2X,73HAND THREE MP00480
     2 DIGITS TO THE RIGHT OF THE DECIMAL POINT ARE PRINTED IN THE MAP ) MP00490
                                                                        MP00500
      PRINT 54,YMAX,YMIN                                                MP00510
   54 FORMAT(/4X,5HYMAX=,E15.7,5X,5HYMIN=,E15.7)                        MP00520
      IF (ICON)5,58,5                                                   MP00530
    5 PRINT 11,BND                                                      MP00540
   11 FORMAT(2X,17HTHE BAND WIDTH IS,E12.5,6H UNITS //4X,14HCONTOUR LEVE MP00550
     1LS          )                                                     MP00560
      I=0                                                               MP00570
      YTOP=AMIN                                                         MP00580
      IF(ABS(YMIN-YMAX)-100.0*BND)53,53,58                              MP00590
   53 YB=YTOP                                                           MP00600
      YTOP=YTOP+BND                                                     MP00610
```

```
      I=I+1                                                             MP00620
      J=MOD(I,20)                                                       MP00630
      ITJZ=IH(J)                                                        MP00640
      IF(YB-YMAX)59,58,58                                               MP00650
   59 PRINT 61,YB,YTOP,ITJZ                                             MP00660
   61 FORMAT(/4X,E10.3,4H TO ,E10.3,2H =,1X,A1)                         MP00670
      GO TO 53                                                          MP00680
   58 NCCP=0                                                            MP00690
      NCP=0                                                             MP00700
   60 PRINT 70                                                          MP00710
   70 FORMAT(1H1)                                                       MP00720
      PRINT 6,T                                                         MP00730
      NLINE=0                                                           MP00740
      NCCP=NCP+1                                                        MP00750
      NCP=NCP+25                                                        MP00760
   73 IF(NCP-M)80,80,75                                                 MP00770
   75 NCP=M                                                             MP00780
   80 J=0                                                               MP00790
      NLINE=NLINE+1                                                     MP00800
      LLINE=N-NLINE+1                                                   MP00810
C SET UP HEADING                                                         MP00820
      IF(NCCP-1) 85,85,90                                               MP00830
   85 J=4                                                               MP00840
   90 DO 100 I=1,135                                                    MP00850
      A(I)=BLK                                                          MP00860
      P(I)=BLK                                                          MP00870
      C(I)=BLK                                                          MP00880
      D(I)=BLK                                                          MP00890
  100 CONTINUE                                                          MP00900
  110 DO 160 L=NCCP,NCP                                                 MP00910
      J=J+3                                                             MP00920
      KI=L                                                              MP00930
      IF(KI-100) 130,120,120                                            MP00940
  120 LL=KI/100                                                         MP00950
      A(J)=KG(LL+1)                                                     MP00960
      KI=KI-100*LL                                                      MP00970
      GO TO 135                                                         MP00980
  130 A(J)=KG(1)                                                        MP00990
  135 J=J+1                                                             MP01000
      IF(KI-10) 150,140,140                                             MP01010
  140 LL=KI/10                                                          MP01020
      A(J)=KG(LL+1)                                                     MP01030
      KI=KI-10*LL                                                       MP01040
      GO TO 155                                                         MP01050
  150 A(J)=KG(1)                                                        MP01060
  155 J=J+1                                                             MP01070
      A(J)=KG(KI+1)                                                     MP01080
  160 CONTINUE                                                          MP01090
```

```
C SETUP FIRST ROW OF ARRAY                                              MP01100
      GO TO 260                                                         MP01110
  170 NLINE=NLINE+1                                                     MP01120
      LLINE=N-NLINE+1                                                   MP01130
      IF(NLINE-N) 180,180,380                                           MP01140
  180 DO 190 I=1,135                                                    MP01150
      A(I)=BLK                                                          MP01160
      B(I)=BLK                                                          MP01170
      C(I)=BLK                                                          MP01180
      D(I)=BLK                                                          MP01190
  190 CONTINUE                                                          MP01200
      IF (ICON)195,260,195                                              MP01210
  195 NCY=NCCP-1                                                        MP01220
      J=1                                                               MP01230
      IF(NCY)200,200,210                                                MP01240
  200 J=5                                                               MP01250
  210 NCY=NCY+1                                                         MP01260
      IF(NCY-NCP) 220,220,260                                           MP01270
  220 IF(NCY-M) 230,260,260                                             MP01280
  230 NLINE = NLINE - 1                                                 MP01290
      YD1=Y(NLINE,NCY)-Y(NLINE+1,NCY)                                   MP01300
      YD2=Y(NLINE,NCY+1)-Y(NLINE+1,NCY+1)                               MP01310
      TP(1)=Y(NLINE,NCY)-0.25*YD1                                       MP01320
      XMT(1)=Y(NLINE,NCY)-0.5*YD1                                       MP01330
      BT(1)=Y(NLINE,NCY)-0.75*YD1                                       MP01340
      TP(5)=Y(NLINE,NCY+1)-0.25*YD2                                     MP01350
      XMT(5)=Y(NLINE,NCY+1)-0.5*YD2                                     MP01360
      BT(5)=Y(NLINE,NCY+1)-0.75*YD2                                     MP01370
      NLINE = NLINE + 1                                                 MP01380
      D1=0.25*(TP(5)-TP(1))                                             MP01390
      D2=0.25*(XMT(5)-XMT(1))                                           MP01400
      D3=0.25*(BT(5)-BT(1))                                             MP01410
      DO 240 I=2,4                                                      MP01420
      TP(I)=TP(I-1)+D1                                                  MP01430
      XMT(I)=XMT(I-1)+D2                                                MP01440
      BT(I)=BT(I-1)+D3                                                  MP01450
  240 CONTINUE                                                          MP01460
      DO 250 I=1,5                                                      MP01470
      J=J+1                                                             MP01480
      I1=MOD(IFIX((TP(I)-AMIN)/BND),20)+1                               MP01490
      I2=MOD(IFIX((XMT(I)-AMIN)/BND),20)+1                              MP01500
      I3=MOD(IFIX((BT(I)-AMIN)/BND),20)+1                               MP01510
      A(J)=IH(I1)                                                       MP01520
      B(J)=IH(I2)                                                       MP01530
      C(J)=IH(I3)                                                       MP01540
  250 CONTINUE                                                          MP01550
      GO TO 210                                                         MP01560
  260 NCY=NCCP-1                                                        MP01570
```

```
      J=0                                                               MP01580
      IF(NCY) 265,265,270                                               MP01590
  265 J=-1                                                              MP01600
      GO TO 330                                                         MP01610
  270 NCY=NCY+1                                                         MP01620
      IF(NCY-NCP) 280,280,310                                           MP01630
  280 J=J+2                                                             MP01640
      THLD=AHLD*Y(NLINE,NCY)+B7                                         MP01650
      IF(THLD) 285,290,290                                              MP01660
  285 D(J)=EMI                                                          MP01670
      GO TO 295                                                         MP01680
  290 D(J)=EPL                                                          MP01690
  295 NUM=INT(ABS(THLD-INT(THLD))*1000.0+0.5)                           MP01700
      NDS=100                                                           MP01710
      DO 300 KK=1,3                                                     MP01720
      J=J+1                                                             MP01730
      KI=NUM/NDS                                                        MP01740
      D(J)=KG(KI+1)                                                     MP01750
      NUM=NUM-KI*NDS                                                    MP01760
      NDS=NDS/10                                                        MP01770
  300 CONTINUE                                                          MP01780
      GO TO 270                                                         MP01790
  310 IF(NCP-M) 360,320,320                                             MP01800
  320 IF(J-127)330,330,360                                              MP01810
  330 J=J+3                                                             MP01820
      KI=NLINE                                                          MP01830
      IF(KI-100) 340,335,335                                            MP01840
  335 LL=KI/100                                                         MP01850
      D(J)=KG(LL+1)                                                     MP01860
      KI=KI-100*LL                                                      MP01870
      GO TO 343                                                         MP01880
  340 D(J)=KG(1)                                                        MP01890
  343 J=J+1                                                             MP01900
      IF(KI-10) 350,345,345                                             MP01910
  345 LL=KI/10                                                          MP01920
      D(J)=KG(LL+1)                                                     MP01930
      KI=KI-10*LL                                                       MP01940
      GO TO 355                                                         MP01950
  350 D(J)=KG(1)                                                        MP01960
  355 J=J+1                                                             MP01970
      D(J)=KG(KI+1)                                                     MP01980
      IF(NCY-1) 270,270,360                                             MP01990
  360 IF(NLINE-1)362,362,368                                            MP02000
  362 PRINT 370,(A(I),I=1,132),(B(IP1),IP1=1,132),(D(IP2),IP2=1,132)    MP02010
      GO TO 170                                                         MP02020
  368 PRINT 370, (A(I),I=1,132),(B(IP1),IP1=1,132),(C(IP2),IP2=1,132),  MP02030
     1(D(IP3),IP3=1,132)                                                MP02040
  370 FORMAT(132A1)                                                     MP02050
```

```
      GO TO 170                                                         MP02060
  380 DO 390 I=1,135                                                    MP02070
      A(I)=BLK                                                          MP02080
      B(I)=BLK                                                          MP02090
      C(I)=BLK                                                          MP02100
      D(I)=BLK                                                          MP02110
  390 CONTINUE                                                          MP02120
      J=0                                                               MP02130
      IF(NCCP-1) 395,395,400                                            MP02140
  395 J=4                                                               MP02150
  400 DO 430 L=NCCP,NCP                                                 MP02160
      J=J+3                                                             MP02170
      KI=L                                                              MP02180
      IF(KI-100) 410,405,405                                            MP02190
  405 LL=KI/100                                                         MP02200
      C(J)=KG(LL+1)                                                     MP02210
      KI=KI-100*LL                                                      MP02220
      GO TO 412                                                         MP02230
  410 C(J)=KG(1)                                                        MP02240
  412 J=J+1                                                             MP02250
      IF(KI-10) 420,415,415                                             MP02260
  415 LL=KI/10                                                          MP02270
      C(J)=KG(LL+1)                                                     MP02280
      KI=KI-10*LL                                                       MP02290
      GO TO 422                                                         MP02300
  420 C(J)=KG(1)                                                        MP02310
  422 J=J+1                                                             MP02320
      C(J)=KG(KI+1)                                                     MP02330
  430 CONTINUE                                                          MP02340
      PRINT 370, (B(IP1),IP1=1,132),(C(IP2),IP2=1,132)                  MP02350
      IF(NCP-M)60,500,500                                               MP02360
  500 RETURN                                                            MP02370
      END                                                               MP02380
```

```
C
C   THIS IS A PROGRAM TO DO AN OBJECTIVE ANALYSIS FOR NSSL NETWORK.
C   X-AXIS POINTS EAST AND Y-AXIS POINTS NORTH.  POINT 1,1 IS LOWER
C   LEFT CORNER OF THE GRID
C
      DIMENSION Z(24,24),ZDUM(24,24),TITLE(24),XTRA(24,24)
C   Z = FIELD TO BE ANALYZED
C   ZDUM = FIELD TO USE IN METMAP
C   TITLE = TITLE TO USE IN METMAP
C   XTRA = FIELD TO STORE ORIGINAL Z-FIELD IF SEVERAL VALUES OF 'C'
C   ARE TO BE TESTED
      DO 975 NSETS=1,2
C   TWO CARDS NEED TO BE CHANGED EACH RUN.  THEY ARE THE PRECEEDING
C   CARD THAT DETERMINES THE NUMBER OF SETS OF DATA THAT ARE TO BE
C   ANALYZED, AND THE WRITE(7,    ) STATEMENT THAT DETERMINES WHAT
C   FORMAT IS TO BE USED ON THE PUNCHED CARDS. FOUR DATA CARDS ARE
C   REQUIRED FOR EACH SET OF DATA TO BE ANALYZED.  THE NINE ZGES
C   VALUES ARE FOR THE 'REGIONS OF INFLUENCE'.  BND AND SCALE ARE
C   FOR CALLING ARGUMENTS FOR SUBROUTINE METMAP.  C = THE ARBITRARY
C   CONSTANT IN THE LAPLACIAN PART OF THE ANALYSIS SCHEME.
      NI=24
      NJ=24
C   NI = NUMBER OF GRID POINTS ALONG THE X-AXIS
C   NJ = NUMBER OF GRID POINTS ALONG THE Y-AXIS
      NI1=NI-1
      NJ1=NJ-1
      READ(5,10) TITLE
   10 FORMAT(20A4)
      READ(5,20)  ZGES1,ZGES2,ZGES3,ZGES4,ZGES5,ZGES6,ZGES7,ZGES8,ZGES9,
     1BND,C,SCALE
   20 FORMAT(6F10.4)
      ZGES=(ZGES1+ZGES2+ZGES3+ZGES4+ZGES5+ZGES6+ZGES7+ZGES8+ZGES9)/9.0
      DO 22 I=1,14
      DO 22 J=1,5
      Z(I,J) = ZGES1
      XTRA(I,J)=Z(I,J)
   22 CONTINUE
      DO 23 I=15,24
      DO 23 J=1,5
      Z(I,J) = ZGES2
      XTRA(I,J)=Z(I,J)
   23 CONTINUE
      DO 24 I=1,6
      DO 24 J=6,11
      Z(I,J) = ZGES3
      XTRA(I,J)=Z(I,J)
   24 CONTINUE
      DO 25 I=7,17
```

```
      DO 25 J=6,11
      Z(I,J) = ZGES4
      XTRA(I,J)=Z(I,J)
   25 CONTINUE
      DO 26 I=18,24
      DO 26 J=6,11
      Z(I,J) = ZGES5
      XTRA(I,J)=Z(I,J)
   26 CONTINUE
      DO 27 I=1,10
      DO 27 J=12,20
      Z(I,J) = ZGES6
      XTRA(I,J)=Z(I,J)
   27 CONTINUE
      DO 28 I=11,17
      DO 28 J=12,20
      Z(I,J) = ZGES7
      XTRA(I,J)=Z(I,J)
   28 CONTINUE
      DO 29 I=18,24
      DO 29 J=12,24
      Z(I,J) = ZGES8
      XTRA(I,J)=Z(I,J)
   29 CONTINUE
      DO 30 I=1,17
      DO 30 J=21,24
      Z(I,J) = ZGES9
      XTRA(I,J)=Z(I,J)
   30 CONTINUE
C  NOW PERFORM THE ANALYSIS SCHEME
      BOUND=Z(10,1)
   70 DO 150 LPASS=1,11
      IF(LPASS.EQ.1) GO TO 124
      DO 110 LPLAC=1,10
      DO 100 I=2,NI1
      DO 100 J=2,NJ1
C
C  THE FOLLOWING 'IF' STATEMENTS ARE NECESSARY SO AS NOT TO ALTER THE
C  VALUES OF THE KNOWN GRID POINTS
C
      IF(I.EQ.10.AND.J.EQ. 1) GO TO 100
      IF(I.EQ.19.AND.J.EQ. 3) GO TO 100
      IF(I.EQ.11.AND.J.EQ. 9) GO TO 100
      IF(I.EQ. 2.AND.J.EQ. 9) GO TO 100
      IF(I.EQ.23.AND.J.EQ. 9) GO TO 100
      IF(I.EQ.15.AND.J.EQ.14) GO TO 100
      IF(I.EQ. 5.AND.J.EQ.17) GO TO 100
      IF(I.EQ.21.AND.J.EQ.18) GO TO 100
```

```
      IF(I.EQ.11.AND.J.EQ.23) GO TO 100
      PL=Z(I+1,J)+Z(I-1,J)+Z(I,J+1)+Z(I,J-1)-4.0*Z(I,J)
      Z(I,J)=Z(I,J)+C*PL
  100 CONTINUE
  110 CONTINUE
C
C  NOW DO THE EIGHT POINT AVERAGE
C
  124 DO 125 J=2,NJ1
      DO 125 I=2,NI1
      IF(I.EQ.10.AND.J.EQ. 1) GO TO 125
      IF(I.EQ.19.AND.J.EQ. 3) GO TO 125
      IF(I.EQ.11.AND.J.EQ. 9) GO TO 125
      IF(I.EQ. 2.AND.J.EQ. 9) GO TO 125
      IF(I.EQ.23.AND.J.EQ. 9) GO TO 125
      IF(I.EQ.15.AND.J.EQ.14) GO TO 125
      IF(I.EQ. 5.AND.J.EQ.17) GO TO 125
      IF(I.EQ.21.AND.J.EQ.18) GO TO 125
      IF(I.EQ.11.AND.J.EQ.23) GO TO 125
      Z(I,J)=(Z(I+1,J)+Z(I-1,J)+Z(I,J+1)+Z(I,J-1)+Z(I+1,J+1)+
     1Z(I-1,J+1)+Z(I-1,J-1)+Z(I+1,J-1))/8.0
  125 CONTINUE
C
C  NOW SET BOUNDARY CONDITIONS TO SAME VALUE AS NEAREST ROW OR COLUMN
C
      DO 135 I=1,NI
      DO 135 J=1,NJ
      Z(1,J)=Z(2,J)
      Z(I,1)=Z(I,2)
      Z(NI,J)=Z(NI-1,J)
      Z(I,NJ)=Z(I,NJ-1)
  135 CONTINUE
      Z(10,1)=BOUND
  150 CONTINUE
C
C  NOW HAVE ANALYZED FIELD
C  NOW SET UP FIELD TO USE IN METMAP
C
  199 DO 250 I=1,NI
      DO 250 J=1,NJ
      ZDUM(J,I)=Z(I,NJ-J+1)
  250 CONTINUE
      CALL METMAP(ZDUM,NI,NJ,TITLE,BND,SCALE,0.0,ZGES,0,1)
  310 FORMAT(12F6.1)
  311 FORMAT(12F6.0)
      C=C+.01
      IF(C.GT..10) GO TO 900
      DO 350 I=1,NI
```

```
      DO 350 J=1,NJ
      Z(I,J)=XTRA(I,J)
  350 CONTINUE
      GO TO 70
  900 WRITE(7,311)  ((Z(I,J),I=1,NI),J=1,NJ)
  975 CONTINUE
  999 STOP
      END
```

INITIAL DISTRIBUTION LIST

		Number of Copies
1.	Lieutenant R. A. Anawalt NROTC Unit Massachusetts Institute of Technology Cambridge, Massachusetts 02139	10
2.	R. L. Alberty Code 51A1 Department of Meteorology Naval Postgraduate School Monterey, California 93940	20
3.	Library Naval Postgraduate School Monterey, California 93940	2
4.	Defense Documentation Center Cameron Station Alexandria, Virginia 22314	20
5.	Naval Weather Service Command Washington Navy Yard Washington, D. C. 20390	1
6.	Officer in Charge Navy Weather Research Facility Naval Air Station, Building R-48 Norfolk, Virginia 23511	2
7.	Commanding Officer Fleet Weather Central COMNAVMARIANAS, Box 12 FPO San Francisco, California 96630	1
8.	Commanding Officer Fleet Weather Central Box 110 FPO San Francisco, California 96610	1
9.	Superintendent Naval Academy Annapolis, Maryland 21402	1
10.	Director, Naval Research Laboratory Attn: Tech. Services Info. Officer Washington, D. C. 20390	1

INITIAL DISTRIBUTION LIST, CONTINUED

		Number of Copies
11.	Department of Meteorology Code 51 Naval Postgraduate School Monterey, California 93940	3
12.	Department of Oceanography Code 58 Naval Postgraduate School Monterey, California 93940	1
13.	Commander, Air Weather Service Military Airlift Command U. S. Air Force Scott Air Force Base, Illinois 62226	2
14.	Atmospheric Sciences Library Environmental Science Service Administration Silver Spring, Maryland 20910	1

Unclassified
Security Classification

DOCUMENT CONTROL DATA - R & D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1 ORIGINATING ACTIVITY (Corporate author) Naval Postgraduate School Monterey, California	2a. REPORT SECURITY CLASSIFICATION Unclassified 2b. GROUP

3 REPORT TITLE

AN APPROACH TO THE NUMERICAL MODELLING OF CUMULUS-SCALE MOTIONS

4 DESCRIPTIVE NOTES (Type of report and inclusive dates)

Master of Science in Meteorology Thesis

5 AUTHOR(S) (First name, middle initial, last name)

Richard Arthur Anawalt

6 REPORT DATE	7a. TOTAL NO. OF PAGES	7b. NO. OF REFS
June 1969	103	12

8a. CONTRACT OR GRANT NO b. PROJECT NO. c. N/A d.	9a. ORIGINATOR'S REPORT NUMBER(S) N/A 9b. OTHER REPORT NO(S) (Any other numbers that may be assigned this report)

10 DISTRIBUTION STATEMENT

This document has been approved for public release and sale; its distribution is unlimited.

11. SUPPLEMENTARY NOTES	12. SPONSORING MILITARY ACTIVITY
N/A	Naval Postgraduate School

13. ABSTRACT

A numerical model which utilizes the isobaric vorticity equation is developed and applied to cumulus-scale data. The model, together with a modified version of the cumulus convection model of Weinstein and Davis, is applied to data obtained from the National Severe Storms Laboratory in Norman, Oklahoma. The calculations yield real time predictions for height, temperature and relative humidity at seven pressure levels, which are then used as input to the cumulus convection model to obtain vertical profiles of various parameters at specified grid points.

Some results of the calculations are presented along with suggestions for further testing and improvement. The results indicate that further modifications to the approach used are necessary in order to provide more accurate forecasts. Values of the individual terms in the vorticity equation are presented as computed from the observed mesoscale data.

DD FORM 1 NOV 65 1473 (PAGE 1)
S/N 0101-807-6811

Unclassified
Security Classification

A-31408

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Cumulus convection						
Numerical prediction						
Mesoscale data						
Cumulus-scale motions						